

UNIVERSITE DE NANCY I

Numero d'enregistrement au CNRS n° AO 8185

* * *

T H E S E

présentée à l'Université de Nancy I pour obtenir le titre de

DOCTEUR INGENIEUR

par

JEAN-PIERRE

MUSSE

Ingénieur I. S. I. N.

* * *

"ETUDE D'UN SYSTEME D'AIDE A LA PROGRAMMATION

DES MACHINES OUTILS A COMMANDE NUMERIQUE

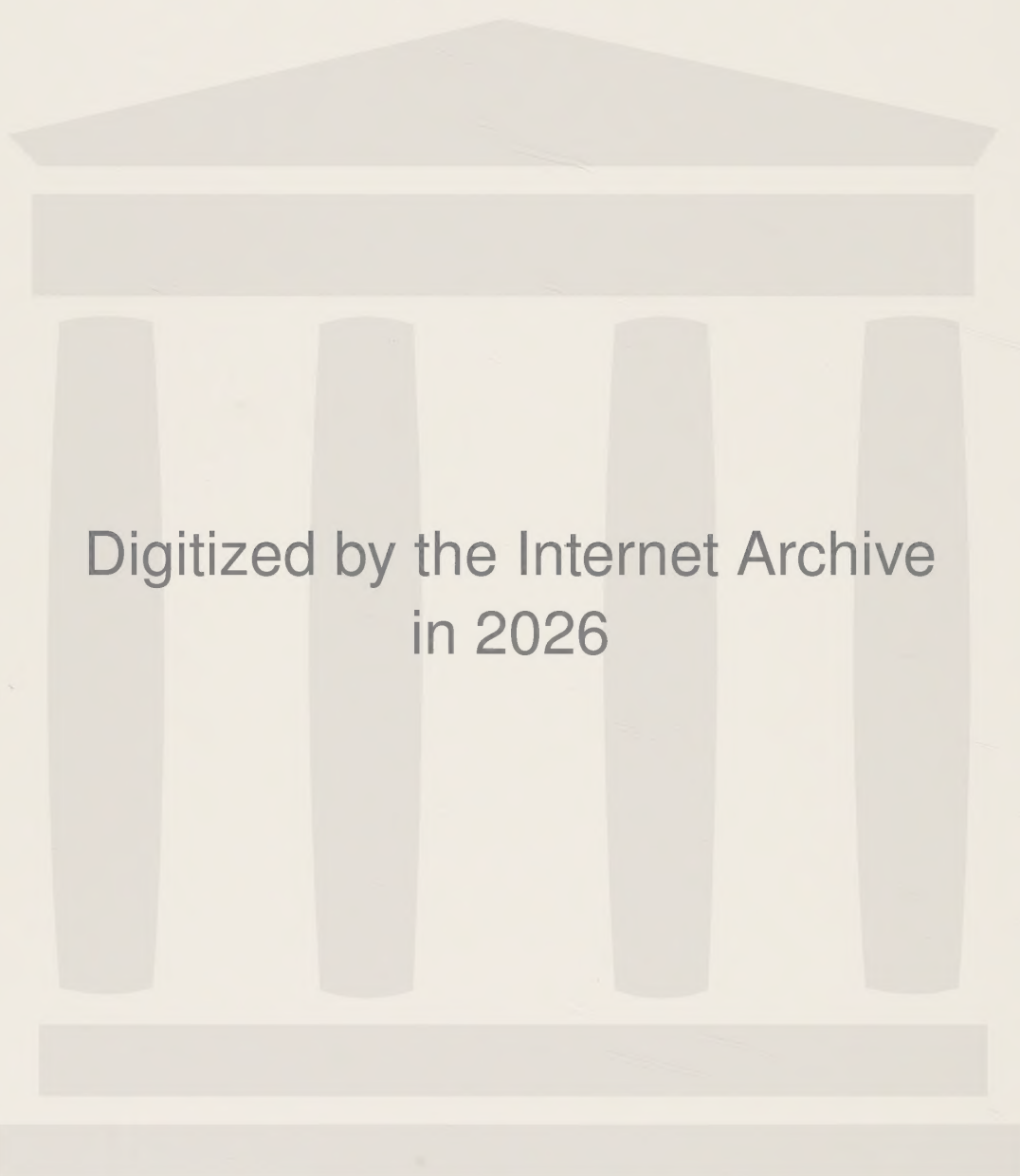
PAR CALCULATEUR INDUSTRIEL"

* * *

soutenue le 9 mars 1973 devant la Commission d'Examen,

Composition du Jury :

Monsieur	C. PAIR	Président
Messieurs	A. FRUHLING	Examineurs
	R. HUSSON	
	J. M. JAMET	
	R. MAINARD	
	M. VERON	



Digitized by the Internet Archive
in 2026

https://archive.org/details/IA41555415_0011

UNIVERSITE DE NANCY I

Numero d'enregistrement au CNRS n° AO 8185

* * *

T H E S E

présentée à l'Université de Nancy I pour obtenir le titre de

DOCTEUR INGENIEUR

par

JEAN-PIERRE

MUSSE

Ingénieur I. S. I. N.

* * *

"ETUDE D'UN SYSTEME D'AIDE A LA PROGRAMMATION

DES MACHINES OUTILS A COMMANDE NUMERIQUE

PAR CALCULATEUR INDUSTRIEL"

* * *

soutenue le 9 mars 1973 devant la Commission d'Examen.

Composition du Jury :

Monsieur	C. PAIR	Président
Messieurs	A. FRUHLING	Examineurs
	R. HUSSON	
	J. M. JAMET	
	R. MAINARD	
	M. VERON	

*
* AVANT-PROPOS *
*

Ce travail a été effectué au Laboratoire d'Electricité et d'Automatique de l'Université de Nancy I , dans le cadre des activités du Centre Universitaire de Commande Numérique animé par Monsieur M. VERON que je tiens à remercier pour le dynamisme qu'il sait faire régner dans son groupe de recherche et l'estime qu'il me manifeste en participant à mon Jury.

Que Monsieur le Professeur A. FRUHLING, Directeur du L. E. A. , trouve ici l'expression de ma gratitude pour ses conseils et l'intérêt bienveillant qu'il m'a toujours témoigné.

Je remercie Monsieur le Professeur C. PAIR, Directeur de l'Institut Universitaire de Calcul Automatique, d'avoir bien voulu examiner mon travail et de me faire l'honneur de présider la commission d'examen.

Mes remerciements vont également à Monsieur J. M JAMET Ingénieur à la Télémécanique Electrique, pour l'intérêt qu'il porte à mon travail et qu'il manifeste en ayant accepté de participer à mon Jury.

Monsieur le Professeur R. MAINARD , Directeur de l'IUT , a toujours soutenu les activités du groupe Commande Numérique et me fait l'honneur de participer à mon Jury ; je l'en remercie très sincèrement.

Je tiens à remercier également Monsieur le Professeur
R. HUSSON d'avoir examiné mon travail et accepté de
siéger à mon Jury.

Mes camarades du Laboratoire et en particulier ceux du
groupe Commande Numérique m'ont apporté une collabora-
tion constructive et sympathique. Mademoiselle J. BOGUET,
secrétaire du Laboratoire, a su présenter parfaitement ce
manuscrit grâce à sa diligente et souriante compétence.
Que tous trouvent ici l'expression de mes remerciements
amicaux.

* *
*

 * TABLE DES MATIERES *

	Pages
INTRODUCTION	
POSITION DU PROBLEME	1
CHAPITRE I : Comparaison du système	
proposé avec les systèmes existants	3
I-1 : Caractéristiques de quelques	
systèmes	3
I-2 : Caractéristiques du système	
proposé	17
I-3 : Avantages de ce système	19
CHAPITRE II : Structure du système proposé	22
II-1 : Le Superviseur SUPMO	23
II-2 : Les processeurs GBS, GBM	
et TRS	26
CHAPITRE III : Notice d'utilisation	32
III-1 : Généralités	32
III-2 : Le superviseur	34
III-3 : Les processeurs TRM et TRS	39
III-4 : Le processeur COB	40
III-5 : Les processeurs GBS et GBM	45
III-6 : Utilisation du système d'aide	
à la programmation	80
EXEMPLE DE PROGRAMMATION	82
CONCLUSION	86
BIBLIOGRAPHIE	87
ANNEXES	
Table des matières des annexes	89

* INTRODUCTION *
*

La programmation manuelle des machines outils à commande numérique est fastidieuse dès que la pièce à usiner devient quelque peu complexe.

Depuis quelques années apparaissent des systèmes d'aide à la programmation qui peuvent être mis en oeuvre sur des calculateurs de bureau ou des "mini-ordinateurs".

Si ces langages ne permettent pas de résoudre tous les problèmes, ils apportent à l'utilisateur une aide efficace dans l'élaboration des programmes des pièces courantes, car ils sont presque tous conversationnels.

Cependant, seules les erreurs de syntaxe et d'écriture sont détectées et peuvent être corrigées, la géométrie de la pièce ne pouvant être vérifiée qu'après l'élaboration complète du programme.

L'objet de ce travail est d'étudier un système d'aide à la programmation capable de détecter les erreurs de syntaxe, mais aussi les erreurs de définitions géométriques par tracé immédiat de la figure définie.

Notre but n'est pas de présenter un système opérationnel, mais de dégager le maximum des possibilités intéressant les utilisateurs pour un système mis en oeuvre sur un ordinateur de petite taille.

Après avoir posé le problème, nous examinerons quelques langages existants ainsi que leurs avantages et inconvénients, puis les caractéristiques du système proposé que nous comparerons aux précédents.

La structure du software et une notice d'utilisation sont donnés dans les chapitres II et III .

* * *
*

POSITION DU PROBLEME

La commande numérique des machines outils est une technique apparue il y a une quinzaine d'années. Elle a été créée pour résoudre un problème d'usinage de pièces aux formes particulièrement complexes et les premières machines fabriquées permettaient d'engendrer des surfaces plus ou moins compliquées. Ce n'est que par la suite que sont apparues des machines plus simples (point-à-point et paraxial).

Mais si cette technique permet de réduire de façon considérable le temps d'usinage, elle implique au niveau du bureau d'étude une préparation et une programmation de l'usinage qui peuvent, dans certains cas, être très longues. C'est pourquoi, il est très vite apparu nécessaire d'avoir recours à l'ordinateur.[1]

Le premier langage de programmation automatique qui est alors apparu (1961) est très puissant : l'APT [2] (Automatically programmed tools). Les autres programmes mis au point par la suite sont eux aussi très élaborés (AUTOPROMPT, SYMPAC, SPLIT ...). Mais tous ces systèmes nécessitant de grands moyens informatiques ne peuvent être utilisés que par de très grosses entreprises. Le télétraitement permet cependant de mettre ces langages de programmation automatique à la portée d'entreprises plus modestes. Toutefois, depuis quelques années, apparaissent des langages de programmation simples, mieux adaptés aux petites entreprises n'ayant pas trop de formes complexes à usiner.

Ces langages sont en général mis en oeuvre, soit sur des ordinateurs de taille modeste, soit sur des calculateurs de bureau auxquels est associée une mémoire auxiliaire. Ils sont presque tous

conversationnels. Leurs possibilités sont bien entendu plus restreintes que celles des gros langages , mais ils permettent toutefois de résoudre la plupart des problèmes d'usinage courant.

Du point de vue de l'utilisation, ces petits langages restent, dans le principe, identiques aux systèmes évolués. La programmation des pièces est réalisée en deux phases

- 1) Définitions géométriques
- 2) Ordres d'usinage

D'autre part, les détections d'erreurs ne portent que sur la syntaxe et le seul moyen de vérifier la géométrie de la pièce est, soit d'usiner un prototype, soit de faire tracer la pièce sur une table traçante à l'aide d'un programme spécialisé.

Or, il paraît intéressant de pouvoir détecter les erreurs de géométrie de la pièce en conversationnel, c'est-à-dire de visualiser pour chaque bloc d'informations la portion de courbe programmée et de vérifier sa conformité. Dans ce cas, un ordre de validation permettrait la prise en compte du bloc d'informations en question.

C'est pourquoi nous proposons un système d'aide à la programmation présentant la possibilité de vérifier la géométrie du profil fini de la pièce tout au long de la programmation de celle-ci.

Or, pour qu'un tel langage soit vraiment simple d'emploi, il est indispensable que les informations géométriques soient confondues avec les ordres d'usinage, la programmation se fait alors en une seule phase.

Nous nous sommes imposé une contrainte supplémentaire : ce programme ne doit pas nécessiter l'utilisation d'une mémoire de masse. En effet, la commande directe des machines-outils par calculateur (C.N.C. : computerized numerical control) évolue actuellement très vite et d'ici quelques années les armoires de commande comporteront pratiquement toutes un "mini-ordinateur" comme élément de base. Celui-ci pourra, à côté des tâches de commande propre de la machine, assurer par exemple une aide à la programmation, moyennant une extension de configuration. C'est dans ce contexte que le système proposé a été élaboré.

Au cours du chapitre I, nous examinerons les caractéristiques de certains langages et systèmes d'aide à la programmation existants, nous décrirons ensuite le système que nous proposons et essayerons de comparer les avantages et inconvénients respectifs.

CHAPITRE I

COMPARAISON DU SYSTEME PROPOSE AVEC LES SYSTEMES EXISTANTS

I-1 Caractéristiques de quelques systèmes :

Nous rappellerons les possibilités des systèmes suivants

AUTOPROG	[3]
ELAN	[4]
KPROPS	[5]

MICROCAP	[6]
PAM	[7][8]
PROFILDATA	[9]
ROMANCE	[10]
PROMO	[11]

Il faut noter que ces langages ne s'appliquent pas tous au même type de machine.

Certains sont réservés au point-à-point, d'autres au contournage plan et d'autres ne s'appliquent qu'à la programmation des pièces de tour.

I-1-1 AUTOPROG

Ce système de programmation automatique a été mis en oeuvre par l'Institut de Recherche pour la MachineOutil et le Travail du Métal à Prague.

Il permet uniquement la programmation des pièces de tour.

Il est basé sur les remarques suivantes :

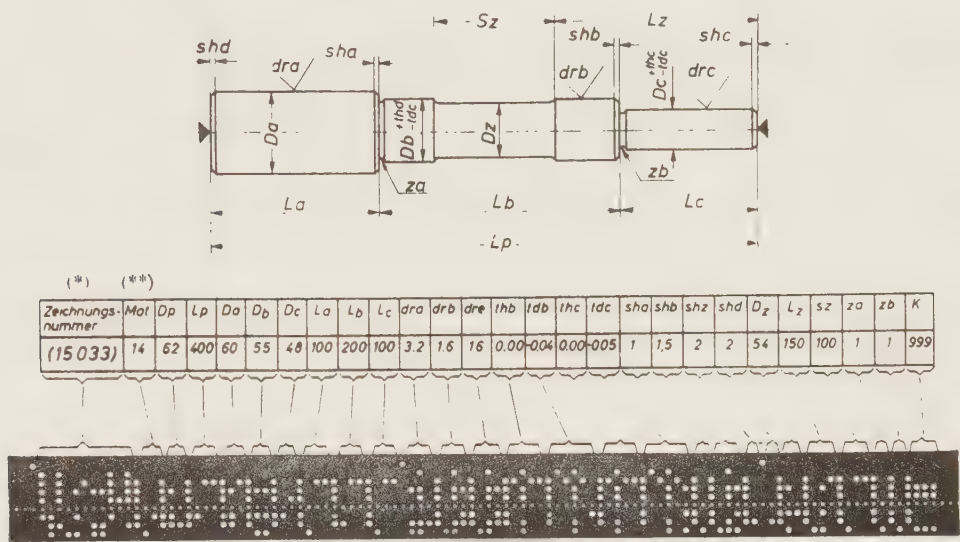
- Seul un langage "hiéroglyphique" bien conçu, pour lequel à chaque mot correspond un symbole, peut être international ;

- Le tournage est l'usinage le plus fréquent dans les petites séries, il représente 30 % au moins des travaux de coupe par enlèvement de métal ;

- Le nombre des types des pièces à tourner est relativement faible, il est donc aisé de les cataloguer.

Ainsi il apparaît que pour chaque famille de pièces il est possible d'escamoter pour ainsi dire le langage de programmation et d'utiliser directement le dessin de la pièce comme "hiéroglyphe" de programmation.

Prenons un exemple. La figure ci-dessous représente une des pièces répertoriées.



Il suffit au programmeur de remplir le tableau des données dont les colonnes successives définissent la matière par un numéro, les longueurs et les diamètres avec indication des tolérances pour certains de ceux-ci, les états de surface, les caractéristiques des plongées et des chanfreins.

La perforation en découle directement comme il est visible sur la figure.

On remarque que ce programme n'utilise aucun mot.

Avantages :

- L'absence de mots facilite la compréhension quel que soit l'utilisateur ;
- Le passage est immédiat du dessin de la pièce, au programme.

Inconvénients :

- Il est nécessaire de disposer d'un répertoire des pièces, donc d'une capacité mémoire non négligeable ;
- Les problèmes d'usinage de pièces non répertoriées ne pourront pas être traités.

I-1-2 ELAN

Ce programme a été mis en oeuvre sur un calculateur Hewlett Pakard modèle 10 (2036 pas de programme, 108 registres numériques) disposant d'un lecteur de cassette magnétique et d'un perforateur de ruban.

Il permet de programmer des pièces définies en point-à-point et se compose de deux parties:

- Un programme général de calcul qui ne tient pas compte de la machine outil utilisée,
- Un programme d'adaptation, spécifique à chaque machine outil.

Le programme général de calcul comporte toutes les instructions habituelles de la programmation manuelle et des instructions de programmation automatique pour l'usinage point-à-point

et paraxial. Il permet :

- la définition des points en coordonnées cartésiennes relatives et polaires ;
- la prise en compte de la longueur d'outil ;
- l'obtention de points sur des figures géométriques avec diverses définitions (points alignés, points sur cercle, grilles, ...) ;
- de répéter après chaque définition de points un ensemble d'instructions.

D'autre part, ce programme détecte les erreurs d'écriture en mode conversationnel.

Le programme d'adaptation, quant à lui, permet :

- de transformer les coordonnées relatives, en coordonnées absolues suivant la machine outil utilisée ;
- de vérifier que les déplacements n'excèdent pas les courses de la machine ;
- de contrôler la validité des fonctions auxiliaires ;
- d'obtenir le ruban perforé définitif ;
- de lire un ruban perforé et de tracer le trajet du centre de l'outil à la même échelle que la pièce à usiner.

Avantages :

- Les moyens de calcul utilisés sont restreints et donc peu coûteux ;
- La mise en oeuvre de ce système est simple.

Inconvénients :

- La vérification de la pièce par tracé n'est faite qu'après l'obtention de la bande perforée ;
- Il n'y a pas de prise en compte du rayon d'outil.

I-1-3 KPROPS

Il s'agit ici d'un programme destiné aux pièces point-à-point, qui peut être mis en oeuvre sur un ordinateur IBM 1130 + 8k + disque.

Il s'agit de la transposition d'un programme plus général : AUTOPROPS.

Il permet de décrire des points

- isolés
- alignés
- disposés en matrice
- disposés en arc de cercle
- résultant d'un ensemble de points par une transformation géométrique telle que

- . translation
- . rotation
- . symétrie
- . homothétie

D'autre part, il est possible de rappeler des groupes de points définis au préalable.

Les instructions définissant les différents points de la pièce sont rédigées dans un langage simple, utilisant des conventions empruntées au FØRTRAN.

Avantages :

- Les instructions géométriques sont nombreuses et simples d'écriture ;
- Ce programme est écrit en FØRTRAN et peut donc ainsi être mis en oeuvre sur n'importe quel type d'ordinateur.

Inconvénients :

- Il est nécessaire de disposer d'une mémoire de masse importante (disque) ;
- Aucune visualisation du profil de la pièce n'est prévue.

I-1-4 MICROCAP

Il s'agit d'un système d'aide à la programmation pour contournage plan qui est implanté sur le minicalculateur Olivetti P 602 avec trois unités périphériques :

- une mémoire auxiliaire à cassette magnétique,
- un perforateur de ruban,
- un lecteur de ruban.

Ce système permet la définition conversationnelle d'instructions géométriques, d'instructions de profil et d'instructions technologiques.

Les instructions géométriques utilisent les propriétés de la géométrie orientée, ce qui permet de lever toutes les ambiguïtés d'une façon élégante.

Les instructions de profil permettent de définir le contour par la succession des éléments géométriques constituant le profil, ainsi que par la nature des intersections des figures.

Les instructions technologiques permettent de définir la cote en Z de la machine, la vitesse d'avance, la tolérance pour l'interpolation circulaire, ainsi que toutes les informations du type programmation manuelle.

Avantages :

- Les moyens de calcul sont peu coûteux car assez restreints ;
- La levée des ambiguïtés est simple à programmer.

Inconvénients :

- L'absence de caractères alphabétiques sur le calculateur P 602 oblige le programmeur à connaître le nombre qui correspond à l'instruction à programmer ;
- Il n'est pas possible de rappeler de façon simple des figures préalablement définies ;
- Il n'est pas prévu de vérification de la géométrie de la pièce.

I-1-5 PAM

Ce "programme d'aide à la programmation manuelle" développé par la société SAPEX permet de résoudre simplement la plupart des usinages sur des machines allant jusqu'à 4 axes.

Il est mis en oeuvre sur un calculateur BURROUGHS Série "L".

Il comporte :

- des instructions manuelles,
- des instructions de calcul ,
 - . instructions arithmétiques
 - . prise en compte des longueurs d'outils
 - . déplacements relatifs
 - . points en coordonnées polaires
- des définitions de points,
 - . points alignés
 - . points sur cercle
 - . points sur grille
- des applications de transformations géométriques ,
 - . rotation
 - . symétrie
 - . translation
- des instructions d'usinage ,
- la possibilité de mémoriser des ensembles de points que l'on peut rappeler au moment opportun,
- des instructions spéciales.

Avantages :

- Possibilité de rappeler des ensembles de points préalablement définis et mémorisés ;
- Utilisation d'un matériel bien souvent déjà présent dans l'entreprise : la machine comptable.

Inconvénients :

- Il n'est pas possible de vérifier la géométrie de la pièce, ni pendant, ni après la programmation de celle-ci ;
- Il n'y a pas de prise en compte des diamètres d'outils pour les usinages paraxiaux.

I-1-6 PROFILDATA

C'est la société FERRANTI qui a élaboré ce langage destiné au contourage plan. Son but essentiel est de rendre le plus rapide possible l'exploitation du dessin de la pièce en fonction de sa cotation usuelle.

Les définitions géométriques sont nombreuses et font souvent double emploi, mais évitent tout calcul et toute interprétation douteuse du dessin.

Le langage est simple, ne faisant pas appel à la connaissance de langues étrangères et évitant d'avoir à écrire des mots dont l'orthographe entraîne souvent des erreurs de format.

Il est possible de programmer, avec l'aide de PROFILDATA, des pièces tridimensionnelles en faisant subir des transformations géométriques simples aux figures définies dans un plan.

Il faut noter qu'il est possible d'associer à l'ordinateur sur lequel est mis en oeuvre PROFILDATA une table à dessiner de précision capable de relever les cotes des points appartenant à un plan ou au tracé d'une pièce.

Ainsi, par exemple, l'ébauche d'une pièce peut se traiter intégralement par la mesure pure, le balayage des points n'ayant pas besoin d'une très grande précision.

Avantages :

- Rapidité de programmation des ébauches grâce à la table à dessiner ;
- Les instructions sont nombreuses et simples d'écriture.

Inconvénients :

- Il est nécessaire de disposer d'un ordinateur assez important ;
- La programmation ne peut pas être effectuée en conversationnel et il n'y a aucun contrôle de la géométrie de la pièce.

I-1-7 ROMANCE

Utilisable sur ordinateur IBM 1130 + 8k + disque, ROMANCE permet la programmation de pièces définies en point-à-point.

Quatre fichiers préparés par l'utilisateur, stockés sur disque et facilement modifiables donne à ROMANCE son efficacité :

- Le fichier outils comporte les caractéristiques fonctionnelles et dimensionnelles de chaque outil ;

- Le fichier "séquence d'opérations" comporte la codification des séquences systématiques d'opérations (par exemple : pointer, percer, tarauder).

- Le fichier "matières" comporte, pour chaque opération, les pourcentages à utiliser pour la matière considérée des vitesses de broches et d'avances fixées par le fichier outil pour une matière de référence.

La programmation d'une pièce est effectuée en deux parties :

- Description géométrique (en coordonnées polaires, rectangulaires, absolues ou relatives) :

- . des obstacles (brides, bossages, ...)
- . des origines successives considérées
- . des points isolés, en ligne, en cercle, en matrice
- . des transformations (rotation, translation, ...) des points définis par les instructions précédentes
- . de fraisages paraxiaux, obliques ou circulaires

- Description de l'usinage qui comporte :

- . l'opération (perçage, fraisage, etc...)
- . l'outil considéré
- . la vitesse d'avance
- . les fonctions auxiliaires

Avantages :

- C'est un programme permettant de définir de façon relativement simple les obstacles ;
- Le programme ROMANCE peut rechercher pour chaque outil un cheminement minimum ;
- ROMANCE peut fournir la représentation graphique des mouvements de l'outil, mais ceci après l'élaboration complète du programme.

Inconvénients :

- Il s'agit d'un programme non conversationnel ;
- Il nécessite une mémoire de masse assez importante.

I-1-8 PROMO

Ce programme utilisé à temps partiel un mini calculateur industriel doté d'une mémoire de 8 k mots de 16 bits et d'un télétype.

Son utilisation ne nécessite ni programme d'adaptation ou de post processeur, ni mémoire de masse.

Il permet de définir :

- des points
- des droites
- des cercles
- des structures de points
- des transformations géométriques
- des cycles d'usinage
- des détourages de cercles

D'autre part, il est possible d'utiliser des opérateurs arithmétiques tels que

- addition, soustraction
- multiplication, division
- sinus, cosinus, tangente, arc tangente
- racine carrée

Il est conversationnel, le dialogue avec l'opérateur se faisant :

- au niveau des ordres d'entrée (corrections et redéfinitions possibles)
- au niveau de la sortie du ruban (corrections, validation ou invalidation des blocs).

Avantages :

- Il s'agit là d'une véritable aide à la programmation manuelle, l'utilisateur étant assisté pas à pas dans l'élaboration des blocs d'informations ;
- Les moyens mis en oeuvre sont restreints et utilisés en temps partiel ;
- Le format des blocs d'informations du ruban perforé objet peut être choisi, ce qui permet le traitement du programme en un seul passage.

Inconvénients :

- Le nombre de définitions géométriques permises simultanément est restreint (seules 150 places sont réservées en mémoire pour conserver celles-ci) ;

- Le contrôle de la géométrie de la pièce n'est pas prévu ;
- Ce programme , dans sa configuration actuelle, n'autorise que la programmation des pièces point-à-point et paraxiales.

1-2 Caractéristiques du système proposé :

Nous ne détaillerons pas dans ce chapitre toutes les possibilités du système, le chapitre "Notice d'utilisation" les exposera ultérieurement.

Précisons seulement que ce système se compose

- d'un langage de programmation,
- d'un programme de correction de bandes symboliques ou "machine",
- d'un programme de tracé de bandes symboliques ou "machine".

Tous ces processeurs sont gérés par un superviseur qui facilite leur appel et leur utilisation.

Le langage de programmation présente les particularités suivantes :

- les instructions de définition des figures géométriques apparaissent dans l'ordre d'usinage et joue donc le rôle des instructions de description de l'usinage ;

- Ce langage est totalement conversationnel et le tracé du profil fini de la pièce est exécuté après chaque définition géométrique, que l'on peut alors valider ou non ;

- Les instructions géométriques permettent de définir le profil fini de la pièce ;

- Il est possible de rappeler des figures ou des groupes de points et de changer leurs conditions d'usinage ou les profondeurs de passe ;

- A partir d'une bande symbolique, on peut générer une nouvelle bande, compte-tenu de modifications à apporter aux conditions technologiques (profondeur de passe, rayon, longueur d'outil, vitesse d'avance, etc...).

Avantages :

- Détection et correction des erreurs géométriques en conversationnel grâce au tracé immédiat ;
- On peut obtenir aussi bien le tracé du profil fini de la pièce que le trajet du centre de l'outil ;
- Le calculateur sur lequel ce système est mis en oeuvre ne dispose pas de mémoire de masse ;
- Obtention rapide d'une nouvelle bande lorsque les conditions technologiques changent.

Inconvénients :

- Les instructions géométriques sont en nombre limité, car il n'est possible de définir une figure qu'à partir de la figure qui la précède et de celle qui la suit ;

- Il n'est pas possible de définir un point en coordonnées polaires.

1-3 Avantages de ce système par rapport aux autres langages :

a) Aide à la programmation :

Tout d'abord, nous proposons ici un système qui n'est pas uniquement un langage puisqu'il est composé d'un superviseur et de cinq processeurs destinés à faciliter le travail du programmeur, aussi bien dans l'écriture du programme, que dans le contrôle et la correction de celui-ci.

b) Contrôle de la géométrie :

Jusqu'à présent, les langages d'aide à la programmation ne permettent pas de vérifier en conversationnel la géométrie de la pièce programmée. Il est cependant possible, lorsque la bande perforée destinée à l'armoire de commande a été totalement élaborée, de tracer la trajectoire du centre de l'outil correspondante, mais il est alors difficile de se rendre compte de l'exactitude de la géométrie.

D'autre part, les erreurs décelées à ce niveau ne pourront être corrigées que par une réédition de la bande perforée.

Par contre, le langage proposé ici permet un contrôle et une correction immédiate de la géométrie puisque, après chaque

instruction, l'élément de figure correspondant est tracé puis peut être validé ou non.

c) Changement des grandeurs technologiques :

Dans les langages existant, toute modification des paramètres technologiques (vitesse d'avance, de rotation, rayon d'outil, ...) peut s'effectuer selon la correction à apporter de deux manières :

- Si la grandeur à changer correspond à un mot de programmation manuelle (vitesse d'avance, rotation, changement d'outil, ...) la correction peut s'effectuer directement sur la bande perforée ;

- Si la correction (rayon, longueur d'outil, profondeur de passe, ...) entraîne une modification des coordonnées, la bande symbolique devra être modifiée puis traitée pour obtenir une nouvelle bande machine.

Le système décrit permet de modifier les grandeurs technologiques grâce à un fichier d'entrée spécialisé : fichier MIT, modifications d'informations technologiques. Une nouvelle bande machine peut alors être éditée compte-tenu des modifications introduites sur ce fichier.

d) Facilité de programmation :

La plupart des systèmes existants nécessitent une programmation en deux étages :

- définition de la géométrie de la pièce,
- description de l'usinage.

Ces deux parties sont combinées dans le langage proposé : la description des figures élémentaires dans l'ordre où elles doivent être exécutées tient lieu d'instructions d'usinages.

e) Configuration réduite de l'ordinateur de traitement :

Le système d'aide à la programmation proposé, mis en oeuvre sur un ordinateur T 2000, occupe environ 7 k mots de 20 bits et ne nécessite aucune mémoire auxiliaire. Ceci s'avère particulièrement intéressant dans le contexte de la commande directe des machines outils (CNC). En effet, disposant de l'ordinateur de commande de la machine, il suffira d'augmenter la taille mémoire centrale de celui-ci et de connecter un téléimprimeur et une table traçante, pour pouvoir disposer d'une aide à la programmation intéressante pour la plupart des usinages courants.

Les systèmes actuels sont mis en oeuvre, soit sur des calculateurs de bureau inaptes à la commande directe, soit sur des ordinateurs disposant d'un disque comme mémoire de masse.

CHAPITRE II

STRUCTURE DU SYSTEME PROPOSE

Pour faciliter la tâche de l'utilisateur, tous les programmes sont en mémoire simultanément et leur appel se fait à partir du clavier du télétype, ce qui permet de n'avoir aucune manipulation à effectuer au pupitre du calculateur.

L'appel des différents programmes (ou processeurs) est réalisé par l'intermédiaire d'un superviseur qui permet en outre de définir les périphériques avec lesquels on désire travailler, ce qui augmente la souplesse d'utilisation. Ce procédé autorise la mise en oeuvre du software sur des systèmes de configurations différentes, sans pour autant nécessiter de modifications des programmes.

Le superviseur (SUPMO) peut être lancé à l'adresse 10 du bloc 0 , ou mieux en appuyant sur la touche "APPEL" du télétype. Il est alors possible d'appeler les processeurs suivants :

- GBS : Génération de bandes symboliques,
- GBM : Génération de bandes en format "machine", c'est-à-dire accepté par l'armoire de commande de la machine outil utilisée,
- TRS : Tracé de la pièce correspondant à une bande symbolique,
- TRM : Tracé de la pièce correspondant à une bande "machine",
- COB : Correction de bandes (aussi bien symboliques qu'en format "machine").

Dans ce chapitre, nous allons examiner successivement les caractéristiques générales du superviseur, puis de chacun des différents processeurs.

Le manuel d'utilisation qui fait l'objet du chapitre 3 donnera plus de détails sur les différentes fonctions de chaque programme et la manière de les utiliser.

II-1 Le Superviseur SUPMO :

Lorsque le superviseur est lancé, soit à partir de l'adresse 10 du bloc 0 , soit en appuyant sur la touche APPEL du télé-imprimeur, il émet sur celui-ci le message DAC ** , ce qui indique au programmeur que le superviseur est prêt à recevoir les commandes d'appel du processeur voulu, et d'attribution des périphériques aux différents fichiers.

Chaque processeur utilise en effet pour ses entrées-sorties un certain nombre de fichiers. Par exemple le programme GBS (génération de bandes symboliques) nécessite en entrée un fichier source (SCE) , un fichier correction (COR) , un fichier de modifications d'informations technologiques (MIT) et en sortie un fichier but (BUT).

A chacun de ces fichiers peut être attribué un périphérique que l'on pourra choisir parmi les organes d'entrée-sortie compatibles avec le fichier en question et le processeur appelé.

Si l'un des fichiers nécessaires au programme appelé n'apparaît pas dans les commandes, le superviseur lui attribuera d'office un périphérique standard.

On trouvera au chapitre 4 (I-2) un tableau donnant, en fonction du processeur choisi, les affectations standards ou permises pour chacun des fichiers relatifs à ce processeur.

Lorsque l'on frappe un mot non conforme, ou que l'on attribue à un fichier un périphérique interdit, le superviseur émet le message " * * * " et attend une nouvelle commande.

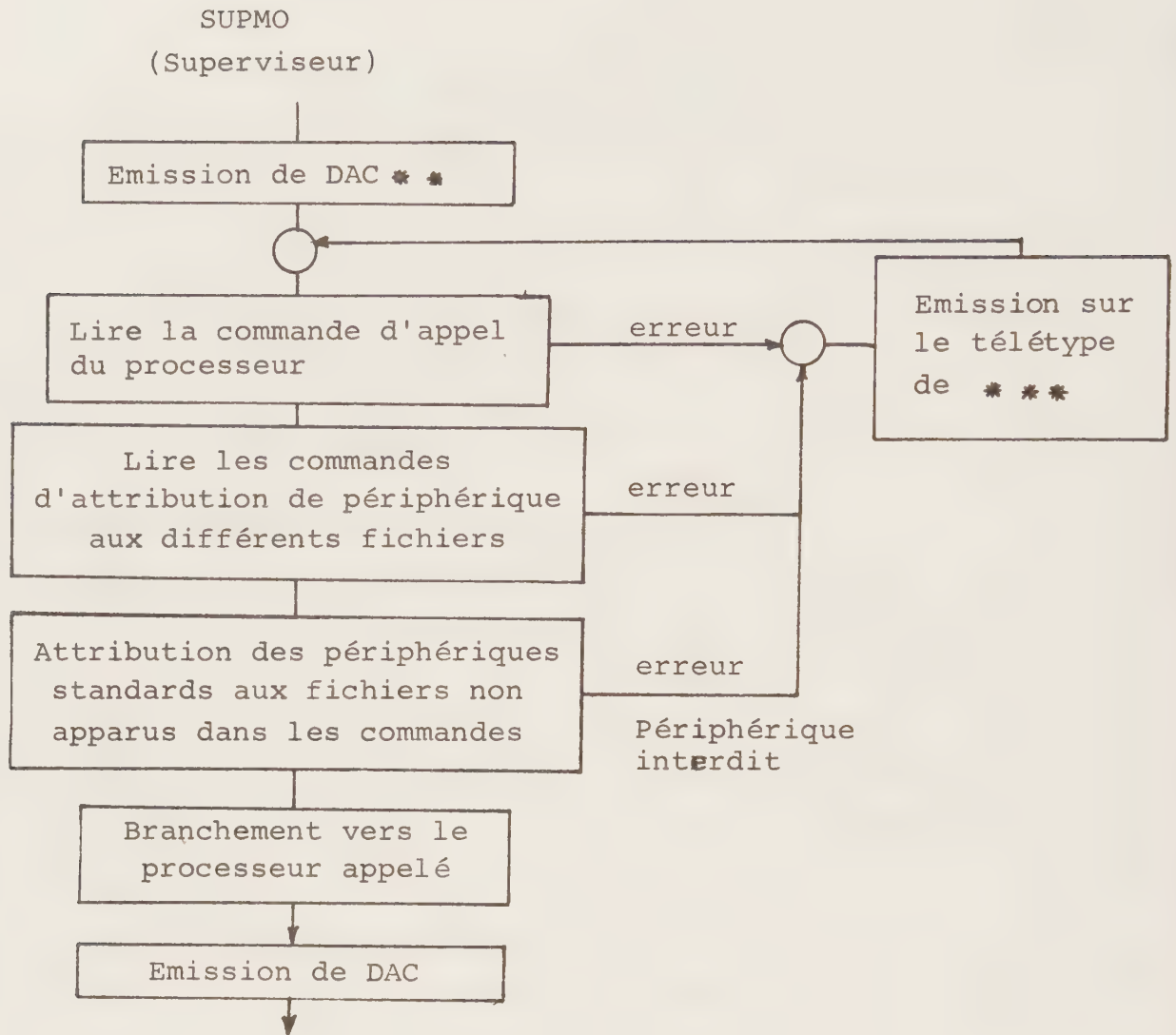
Il est permis de donner plusieurs fois une affectation pour un même fichier, étant entendu que ce sera le dernier périphérique mentionné qui sera pris en compte.

Le superviseur lit les commandes en séquence et les traite aussitôt, ce qui permet de détecter les erreurs d'écriture avant la fin de la phrase.

Les commandes sont des mots de 2 ou 3 lettres et doivent apparaître comme suit :

- Le nom du processeur désiré (3 lettres)
- Le nom du fichier auquel on veut attribuer un périphérique autre que le périphérique standard (3 lettres)
- Le nom du périphérique désiré pour le fichier précédent (2 lettres).

Les espaces sont ignorés mais ne doivent pas apparaître au milieu d'un mot. La phrase se termine par un ";" . Le superviseur teste alors si les périphériques demandés sont permis pour le processeur appelé, et attribue les périphériques standards aux fichiers non apparus. Puis il se branche au début du programme désiré et émet DAC sur le télétype pour indiquer que le processeur est prêt.



II-2 Les processeurs GBS , GBM et TRS :

1) GBS (Génération de bande symbolique)

Ce programme est utilisé normalement avec le téléimprimeur et le perforateur rapide. Il permet alors de générer une bande symbolique à partir d'un programme frappé au clavier. Toutes les erreurs d'écriture et de syntaxe sont détectées et peuvent être corrigées aussitôt. D'autre part, il est possible d'exécuter sur la table traçante la figure correspondant à chaque bloc d'informations avant que celui-ci ne soit perforé, puis de le valider si le tracé convient. Ainsi, la bande générée est exempte de toute erreur de syntaxe et doit correspondre à la pièce à usiner, tout au moins pour ce qui est du profil fini de celle-ci. En effet, il n'est pas possible de détecter les erreurs correspondant à un mauvais positionnement de l'outil par rapport au profil fini.

Ce processeur GBS peut être aussi utilisé avec le lecteur du télétype ou le lecteur rapide comme périphérique source. Il faut alors disposer d'une bande préalablement tapée sur un téléimprimeur "off line". Les erreurs seront détectées au fur et à mesure de la lecture et un message d'erreur émis sur le télétype. Ce message précisera la section et le numéro du bloc dans lequel se situe l'erreur, qu'il est alors possible de corriger à partir du clavier. Chaque bloc devra être déterminé par un caractère de validation afin que la bande se déroule en continu.

2) GBM (Génération de bande "'machine'")

Ce processeur permet à partir d'une bande symbolique de générer une bande compatible avec l'armoire de commande de la

machine outil. Une bande source générée par GBS est exempte de fautes, par contre si elle a été perforée "off line", elle peut comporter des erreurs qui seront détectées par GBM et pourront être alors corrigées. Le tracé du profil fini de la pièce peut aussi être obtenu.

Il est permis également d'utiliser GBM en frappant le programme symbolique au clavier. Dans ce mode d'utilisation, il existe certaines restrictions quant aux phrases d'informations technologiques.

3) TRS (Tracé de bandes symboliques)

Le but de ce processeur est de tracer le profil fini de la pièce programmée à partir d'une bande perforée en symbolique.

En fait, ces trois processeurs correspondent au même programme dont nous donnons l'organigramme général, que nous allons examiner.

2 types de phrases peuvent se présenter :

- les phrases d'informations technologiques qui apparaissent en débit de section
- les phrases d'informations géométriques.

a) Les informations technologiques :

Au début de chaque section doit apparaître une phrase d'informations technologiques.

Celle-ci est composée d'un certain nombre de mots définissant soit les valeurs numériques des paramètres technologiques soit des fonctions particulières.

La lecture de la phrase est effectuée de la manière suivante :

Un caractère est lu puis comparé aux caractères contenus dans un tableau à chaque mot duquel correspond l'adresse du sous-programme qui lui est propre.

Chacun de ces sous-programmes assure la lecture d'un ou plusieurs caractères selon une structure arborescente et initialise les paramètres technologiques apparus dont les valeurs sont conservées durant toute la section. Le point virgule marquera la fin de la phrase.

b) Les informations géométriques :

Leur lecture s'effectue de la même manière que pour la phrase d'informations technologiques. Chaque caractère lu crée un branchement vers un sous-programme dont la structure arborescente assure la lecture complète du mot concerné et le traitement correspondant. Le point virgule marque la fin de la phrase.

Les informations géométriques sont mémorisées et transitent dans quatre blocs différents.

- Le bloc 3 (XP) constitue l'ensemble des mémoires où sont stockées provisoirement les informations lues.

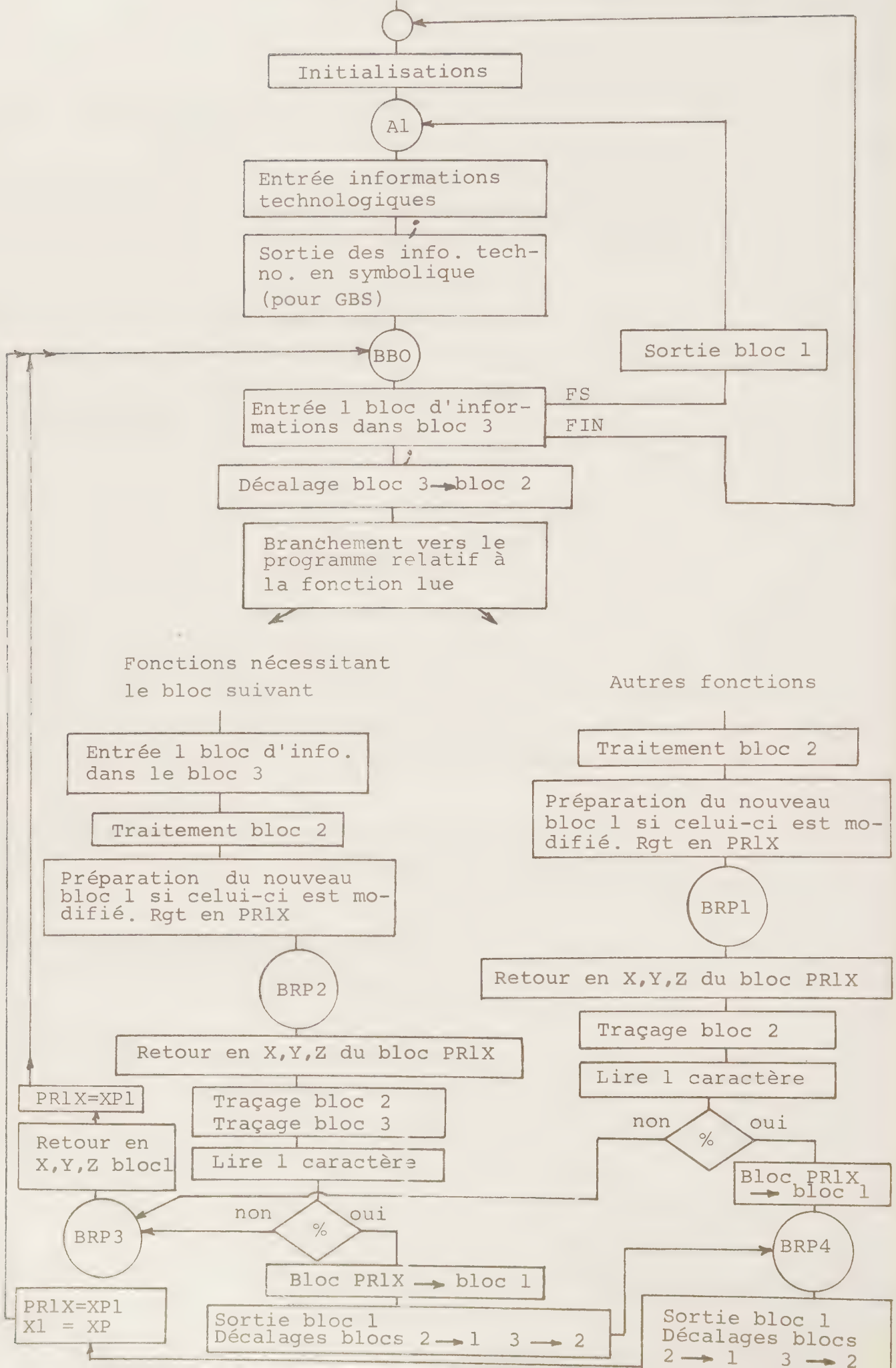
- Dans le bloc 2 (XP2) est mémorisée la fonction traitée par les sous-programmes de calcul géométrique.

- Le bloc 1 (XP1) contient les informations validées prêtes à être perforées.

- le bloc (PR1X) "préparation du bloc 1" est l'image du bloc 1 ,

Programme relatif aux processeurs
GBS, GBM, TRS

29



avec éventuellement des modifications apportées par la fonction apparaissant dans le bloc 2.

Le transit des informations géométriques est effectué comme le montre l'organigramme.

Il est nécessaire d'utiliser un bloc auxiliaire PR1X pour ne pas modifier le bloc 1 tant que la fonction apparaissant dans le bloc 2 n'a pas été validée.

La différence entre les processeurs GBS , GBM et TRS se situe dans le sous-programme de sortie du bloc 1. Ce sous-programme pour TRS sera réduit à un branchement à l'instruction en séquence.

Pour GBS et GBM nous trouverons en annexe les organigrammes plus détaillés du sous-programme de sortie.

Par souci de clarté, nous n'avons pas fait apparaître dans l'organigramme général l'entrée des informations technologiques sur le fichier MIT, ni le retour en début de section, dans le cas où la phrase de données technologiques comporte plusieurs groupes d'informations séparés par une virgule.

Le principe adopté pour la prise en compte des informations du fichier MIT est précisé en annexe, où l'on examinera plus particulièrement les sous-programmes PV , VIR , FS et le programme PRGPR.

II - 3 Processeurs TRM et COB :

Ces deux programmes ne présentant pas de difficultés majeures dans leur structure, nous n'avons pas établi d'organigramme général.

On trouvera en annexe les organigrammes détaillés les concernant.

On peut simplement remarquer que le processeur COB est absolument indépendant, mais que le processeur TRM fait appel au sous-programme de traçage d'un bloc (TRBLO) qui est utilisé par les programmes GBS et GBM.

CHAPITRE III

NOTICE D'UTILISATION -

Le système d'aide à la programmation des machines outils à commande numérique se compose d'un superviseur (SUPMO) et de divers processeurs (GBS, GBM, TRS, TRM, COB).

Nous allons examiner le mode d'utilisation du superviseur puis de chacun des processeurs.

III - 1 Généralités :

Conventions d'écriture

Le tableau ci-dessous précise la signification des abréviations utilisées dans cette notice :

Caractère espace	␣
Caractère line feed	LF
Caractère Retour chariot	RC

Nous utiliserons par la suite les expressions format "machine", langage "machine", ou bandes "machines", pour parler des programmes écrits ou des bandes perforées dans le format accepté par l'armoire de commande de la machine outil (programmation manuelle).

Dans les exemples qui seront donnés ici, les mots émis par les divers processeurs seront soulignés, alors que les informations données par le programmeur ne le seront pas.

Ecriture des nombres

Pour écrire un nombre, il n'est pas nécessaire d'indiquer les zéros qui précèdent celui-ci.

Un nombre doit toujours se terminer par un espace. Une erreur dans l'écriture d'un nombre entraînera l'impression du message d'erreur ERR ECR NB (Erreur d'écriture d'un nombre).

Afin d'éviter certaines erreurs de cadrage, les données d'entrée ne devront pas dépasser un certain nombre de chiffres, celui-ci dépendant de la grandeur programmée. Ainsi les coordonnées ne devront pas avoir plus de sept chiffres, la vitesse d'avance plus de quatre, etc...

Un nombre composé de trop de chiffres sera aussitôt détecté et le message d'erreur NB TR GR (nombre trop grand) sera imprimé.

Correction des erreurs

Lorsque le fichier source est associé à un périphérique autre que le clavier du télétype, la correction des erreurs est tout de même possible. Lorsqu'une erreur est détectée sur la bande source, un message est imprimé, composé du numéro de la section en cours et du numéro de bloc, suivi d'une identification de l'erreur. Il est alors possible de corriger celle-ci à partir du clavier du télétype, en réécrivant correctement le mot erroné ainsi que tout le reste du bloc jusqu'au point virgule inclu. Ensuite, pour relancer la lecture

de la bande source, il faudra frapper le caractère "apostrophe".

Pour pouvoir identifier correctement le bloc où se situe l'erreur, ce mode d'utilisation nécessite de donner un numéro à chaque bloc d'informations.

III-2 Le superviseur SUPMO :

Pour lancer le système, il faut initialiser le pointeur à l'adresse 10 du bloc 0 , puis appuyer sur MARCHE.

Ensuite, aucune manipulation au pupitre n'est utile; en effet, quel que soit le programme en cours, le contrôle du superviseur s'obtient en appuyant sur la touche APPEL du télé-imprimeur.

Quand le superviseur est prêt à recevoir des commandes, il imprime le message DAC** .

III-2 -1 Commandes du superviseur

La phrase de commande se compose d'un groupe de mots de 2 ou 3 caractères. Elle se termine par un point virgule " ; "

- Il faut d'abord taper la commande d'appel du processeur voulu :

GBS Génération de bandes symboliques

GBM Génération de bandes "machine"

TRS Tracé de bandes symboliques

TRM Tracé de bandes "machine "

COB Correction de bandes

- On donnera ensuite les commandes d'affectation de périphériques aux différents fichiers utilisés par le processeur appelé .

Exemple :

DAC** GBS SCE LR BUT TP ;

Ces commandes d'affectation sont composées tout d'abord d'un mot de 3 caractères désignant le fichier :

SCE Fichier source

BUT Fichier but

QOR Fichier correction

MIT Fichier modifications d'informations technologiques

puis d'un mot de 2 lettres désignant le périphérique :

TC Télétype clavier

TP Télétype perforateur

TF Télétype feuille

TL Télétype lecteur

LR Lecteur rapide

PR Perforateur rapide

Les espaces entre les mots ne servent qu'à clarifier la phrase de commande, et ne sont pas pris en compte, ainsi la phrase suivante est correcte :

DAC** GBSSCF TC BUT TP ;

III-2 -2 Affectations des périphériques aux fichiers

Chaque processeur dispose d'une affectation de périphériques standard. D'autre part, les divers fichiers n'admettent que des périphériques qui leur sont propres.

Le tableau ci-dessous indique les affectations possibles des périphériques aux fichiers nécessaires à chaque processeur. Le périphérique standard est souligné

GBS	<u>TC</u> , LR , TL	<u>PR</u> , TP	<u>TC</u>	TC , TL
GBM	<u>LR</u> , TL , TC	<u>PR</u> , TP	<u>TC</u>	TC , TL
TRS	<u>LR</u> , TL			
TRM	<u>LR</u> , TL			
COB	<u>LR</u> , TL	<u>PR</u> , TP	<u>TC</u> , TL	

Si l'on désire travailler avec un processeur en utilisant ses affectations standards, il suffit de taper uniquement la commande d'appel du processeur. Sinon, il suffira de frapper uniquement les commandes d'affectation des fichiers pour lesquels on désire utiliser un autre périphérique.

Ainsi DAC ** GBS ; sera équivalent à

DAC ** GBS SCE TC BUT TP COR TC ;

et DAC ** GBS SCE LR ; sera équivalent à

DAC ** GBS SCE LR BUT TP COR TC ;

III-2-3 Erreurs

Si, avant le point virgule, on a commis une erreur d'affectation, il suffit de retaper la commande correspondante avec le nouveau périphérique choisi. Si le point virgule est déjà frappé, il faut se remettre sous le contrôle du superviseur, en appuyant sur la touche APPEL du téléimprimeur, et refrapper entièrement la phrase de commande.

Un mot erroné en cours de commande entraînera l'impression de *** , la phrase de commande complète doit alors être retapée.

L'impression de *** après le point virgule indique que l'on a affecté un mauvais périphérique à un fichier, d'où la nécessité de refrapper toutes les commandes.

III-2-4 Particularités relatives aux divers processeurs

1) Pour les processeurs GBS et GBM il existe un fichier MIT (modifications des informations technologiques) pour lequel il n'y a

pas d'affectation standard. En effet, le fait de donner une affectation à ce fichier d'entrée indique au processeur GBS ou GBM qu'il faut lire les informations technologiques à modifier, non pas sur le fichier source, mais sur le fichier MIT. Comme nous le verrons plus tard, il existe donc deux modes d'utilisation de GBM ou GBS, suivant que le fichier MIT est appelé ou non.

Si au cours de la phrase de commande, le fichier MIT est appelé par mégarde, on doit, pour le supprimer, réinitialiser le superviseur en appuyant sur la touche APPEL du téléimprimeur.

2) Pour les processeurs GBS, GBM et COB, dès la fin de la phrase de commande, lorsqu'elle est exacte, le superviseur émet le message DAC qui indique que le processeur est prêt.

Par contre, pour TRS et TRM, après le ";" , le superviseur attend de nouvelles informations nécessaires au tracé de la pièce programmée.

Il faut tout d'abord indiquer l'échelle par :

EC
 nombre de 3 chiffres maximum

Le nombre qui suit EC indique combien de $1/100^{\text{ème}}$ de millimètre représente 1 pas de $1/10^{\text{ème}}$ de la table traçante, ainsi EC10 équivaut à une échelle 1, EC100 à une échelle $1/10$, et EC1 à une échelle 10. L'échelle peut donc varier de $1/10^{\text{ème}}$ à 99 .

On précise ensuite la vue désirée :

TT

2 caractères pris parmi X, Y, Z

Le premier caractère correspond au premier axe de la table traçante, le second au deuxième axe.

Toutes les combinaisons sont permises, sauf la répétition du même caractère.

On peut faire apparaître sur le tracé les déplacements rapides en tapant la commande VR.

Ces commandes pourront être répétées dans la même phrase, seules les dernières seront prises en compte. La phrase se termine par un ";" et le message DAC est imprimé.

Exemple de commandes d'appel de TRS :

DAC ** TRS SCE TL ; EC 10 TTXY VR ; DAC

Les erreurs d'écriture dans la deuxième phrase entraînent des messages d'erreur différents suivant la faute commise. La correction est apportée en refrappant uniquement le mot erroné ou l'information manquante.

III-3 Les processeurs TRM et TRS

Les informations contenues dans la phrase d'appel suffisent à ces processeurs. Avant de les lancer par la commande "GØ", la

bande dont on veut le tracé doit être mise en place sur le périphérique source choisi.

Si les informations d'échelle ou de vue ne conviennent plus au cours du tracé, on doit se remettre sous contrôle du superviseur et redonner toutes les commandes nécessaires.

Ces processeurs détectent également les erreurs de syntaxe ou d'écriture. De plus, le processeur TRS indique les grandeurs que l'on a omis de définir. Le tracé s'arrête avec le mot FIN pour TRS et sur le caractère % pour TRM.

Pour exécuter un nouveau tracé, il suffit de placer la bande correspondante sur le lecteur et de frapper la commande GØ.

Il n'est pas possible de commencer la lecture d'une bande ailleurs qu'en son début, ni de modifier l'échelle ou la vue au cours du tracé, sans refaire appel au superviseur.

III-4 Le processeur COB :

Ce processeur permet de recopier et de corriger des bandes perforées aussi bien en symbolique qu'en format 'machine'.

Son utilisation repose sur trois commandes :

- Recopie
- Suppression
- Insertion

III-4-1 Commandes C et S

L'ordre de recopie se présente sous la forme suivante :

C ----- ↑

5 caractères au maximum

L'ordre de suppression sous la forme suivante :

S ----- ↑

5 caractères au maximum

La commande de recopie (de même que celle de suppression) doit indiquer au processeur COB jusqu'où il doit recopier (ou supprimer) la bande source. Pour ce faire, le programme COB arrêtera la lecture dès qu'il rencontrera un groupe de caractères identiques aux caractères indiqués entre le C (ou le S) et le caractère ↑ des commandes de recopie (ou de suppression), le groupe correspondant étant inclus dans la recopie ou la suppression.

Exemple :

Supposons que la bande source se présente sous la forme suivante, à l'endroit où l'on désire apporter une correction :

N151 ▯ G54 ▯ X0005430 ▯ Y0042000

Si l'on désire changer totalement ce bloc d'informations, il faudra arrêter la recopie de la bande juste après N151. On frappera donc la commande CN151↑ qui aura pour effet de recopier la bande jusqu'à N151 compris.

Si seul un ou plusieurs caractères sont à changer il est possible de recopier la bande jusqu'à ce ou ces caractères. Supposons que l'on désire changer le caractère Y pour le transformer en Z , il suffit de taper la commande C 5430␣↑ .

Bien entendu, ces 5 caractères (5430␣) ne doivent pas se trouver précédemment sur la bande source.

Pour éviter toute confusion, en cas de doute, on pourra procéder ainsi : frapper la commande CN151↑ , puis ensuite la commande C 5430␣↑ ou mieux C30␣↑ puisque cette combinaison n'apparaît qu'une seule fois entre N151 et Y .

Lorsque la recopie ou la suppression de la bande source est terminée, le programme COB imprime sur le télétype les 14 derniers caractères lus, afin que le programmeur puisse situer exactement l'endroit où est arrêtée la lecture de la bande source.

Remarque :

Une erreur dans les commandes C ou S peut être corrigée tant que le caractère ↑ n'est pas frappé, on le remplace par ← ce qui annule toute la commande. En cas de commande erronée, ne correspondant à aucun groupe de caractères rencontré sur la bande, la recopie ou la suppression porte sur la bande complète et peut être interrompue en redonnant le contrôle au superviseur.

Exemple :

CN 211← CN 212↑ sera équivalent à CN 212↑

III-4-2 Commande d'insertion

Lorsque l'on désire apporter une correction, il faut indiquer au programme COB que les informations à perforer sur la bande objet sont à prendre non pas sur la bande source, mais sur le fichier COR.

Les corrections ou insertions désirées de 40 caractères au plus doivent être frappées entre deux caractères apostrophes ' ' ', dont le dernier provoque leur perforation sur la bande.

Une modification de la phrase avant la frappe de l'apostrophe final peut être apportée par le caractère ← qui supprime toute la phrase jusqu'à la commande de début d'insertion.

III-4-3 Exemples d'utilisation de COB

Soit à corriger les blocs d'information suivants

N 150	↵	X0072000	↵	Y0006000
N 151	↵	X005430	↵	Y0042000

Dans le bloc N151, il manque l'information G54 et le caractère Y est erroné et doit être remplacé par Z.

La succession des commandes sera la suivante :

CN 151 ↑ (copie de la bande jusqu'aux caractères N151 compris)

Y0006000

N151

"G54" C 30↵ (insertion de G54 et copie de la bande jusqu'aux caractères 30 ↵ compris)

N 151 X005430

SY↑

(suppression du caractère Y)

N 151 X005430 Y

'Z'

CN 170

(copie de la bande jusqu'à la prochaine
correction)

III-4-4 Messages d'erreur relatifs à COB

Une erreur dans l'écriture d'une commande entraîne un des messages suivants :

*** Une commande est erronée : frappe d'un caractère autre que C , S, ;,espace , Retour tête, line feed. Une commande C ou S comporte un groupe de plus de 5 caractères.

? La phrase de correction ou d'insertion est trop longue. Frapper un caractère ' si la phrase est correcte, ou ← si la phrase est erronée.

Pour continuer la correction, taper un nouveau caractère ' et continuer l'écriture des informations.

Ce message d'erreur peut également se présenter, si après une phrase d'insertion, on omet de frapper le caractère ', ce qui aura pour effet de continuer à prendre les informations frappées comme des corrections et non comme des commandes.

III-4-5 Limites d'utilisation de COB

Lorsque l'on se sert du processeur COB avec le perforateur rapide affecté au fichier BUT, il n'y a aucune restriction dans son utilisation.

Par contre, si le perforateur du télétype est affecté au fichier BUT, il n'est pas possible de corriger des bandes en code "machine". En effet, la plupart des armoires de commandes n'acceptent pas le caractère "Perfo off" qui se trouvera sur les bandes corrigées.

D'autre part, pour la correction de bandes symboliques avec le perforateur du télétype, il faut toujours arrêter la lecture à la fin d'un mot, en effet si on ne prend pas ce soin le caractère "perfo off" apparaîtrait au milieu d'un mot et créerait une erreur d'écriture.

III-5 Les processeurs GBS et GBM :

III-5-1 Généralités

Ces deux processeurs GBS (génération de bandes symboliques) et GBM (génération de bandes machines) admettent en entrée le même langage symbolique. GBS fournit en sortie une bande en langage symbolique et GBM une bande assimilable par une armoire de commande de Machine Outil.

Ce programme d'entrée écrit en langage symbolique est divisé en sections composées d'un ensemble de blocs d'informations géométriques précédé d'une phrase d'informations technologiques.

Chaque phrase d'exécution correspondant à un outil particulier ou à des conditions de coupe différentes, doit être traduite dans une nouvelle section.

Les informations technologiques permettent de définir les grandeurs nécessaires à l'usinage de la pièce (rayon et longueur d'outil, cycles particuliers, profondeur de passe, vitesse d'avance, etc...).

Les informations géométriques définissent le profil de la pièce finie, grâce à un certain nombre de fonctions que nous étudierons ultérieurement.

III-5-2 Informations technologiques

Les grandeurs définies dans la phrase d'informations technologiques sont indispensables pour la génération d'une bande en code machine (correction de rayon et de longueur d'outil, programmation des cycles, etc...) , grandeurs que seul utilise le processeur GBM, GBS se limitant à les transcrire sous format standard.

Il peut être utile de disposer de plusieurs groupes d'informations technologiques pour une même section, par exemple perçage, puis taraudage d'un même groupe de trous.

Aussi la phrase de définition des grandeurs technologiques peut être composée de plusieurs sous-ensembles qui seront traités successivement par GBM.

La phrase est composée de 3 sortes de mots :

- grandeurs : code de 1 à 3 caractères suivi d'un nombre de n chiffres , n dépendant de la grandeur programmée.

Exemple :

VA 2300

R 10 000

Le nombre peut être séparé du code par un ou plusieurs espaces et doit toujours se terminer par un espace (voir chapitre I-2).

- commandes : code de 2 ou 3 caractères suivi d'un espace.

Exemple :

ALA

NAR

- séparateurs : caractères, " ; " , "

Les différents mots pouvant composer la phrase d'informations technologiques sont :

- SE nombre de 2 chiffres

Indique le numéro de la section.

- TT suivi ou non de 2 caractères pris parmi X, Y ou Z

Indique que l'on désire un tracé, les 2 caractères suivant TT précisent la vue choisie.

- TS supprime le tracé.

- EC nombre de 3 chiffres

Précise l'échelle du dessin désiré (voir III-2-4).

- R nombre de 7 chiffres

Exprime le rayon de l'outil en microns.

- L nombre de 7 chiffres
Exprime la longueur de l'outil en microns.

- PR nombre de 7 chiffres
Indique la profondeur de passe (suivant le rayon)
Cette grandeur exprimée en microns traduit la distance entre le profil fini et le profil à usiner pour la passe en question.

- PL nombre de 7 chiffres
Grandeur qui indique la profondeur de passe (suivant la longueur).
Elle est exprimée en microns et représente la distance entre le profil fini et celui à usiner.

- VA nombre de 4 chiffres
Vitesse d'avance en millimètres par minute.

- VR — nombre de 4 chiffres
1 caractère P ou N
Vitesse de rotation de broche exprimée en tours par minute. La lettre P précise une rotation dans le sens positif, la lettre N dans le sens négatif.

- AR Commande de mise en route de l'arrosage.

- NAR Commande d'arrêt de l'arrosage.

- GA Précise le côté d'usinage . Le centre de la fraise est à gauche du profil dans le sens du parcours.

- DR Le centre de la fraise est à droite du profil.
- SC Précise qu'il n'y a pas de correction d'outil.
- TA Cycle de taraudage.
- PEN Cycle de perçage normal.
- PEP Cycle de perçage de trous profonds.
- PEA Cycle de perçage avec arrêt en fin de perçage (lamage).
- ALA Cycle d'alésage avec arrêt de la broche avant la remontée en vitesse travail.
- ALT Cycle d'alésage avec remontée de la broche en vitesse travail.
- ALR Cycle d'alésage avec remontée de la broche en vitesse rapide (après arrêt).
- ZD nombre de 7 chiffres
Précise la cote de déplacement en vitesse rapide pour les différents cycles.
- ZP nombre de 7 chiffres
Précise la cote de perçage, alésage ou taraudage.
- , Séparateur des groupes d'informations technologiques.
- ; Fin de la phrase de données technologiques.
- < Commentaires >
Caractères de début et fin de commentaires.

- CO Changement d'outil.

Une phrase d'informations technologiques comportera un certain nombre de ces grandeurs et commandes. Celles-ci peuvent apparaître dans un ordre absolument quelconque et peuvent être répétées, seule la dernière étant prise en compte. Elles peuvent être ou non séparées par un ou plusieurs espaces, retour tête ou line feed.

Le premier groupe d'informations technologiques apparaissant dans le programme doit être complet, les groupes suivants ne précisant que les modifications par rapport aux informations précédentes. Ceci est valable également pour les phrases d'informations technologiques des sections successives.

Deux types d'informations technologiques peuvent se présenter :

1) Les informations relatives à un fraisage qui devront préciser :

- le numéro de section	SE
- le rayon d'outil	R
- la longueur d'outil	L
- la vitesse d'avance	VA
- la vitesse de rotation	VR $\begin{matrix} P \\ N \end{matrix}$

On peut d'autre part indiquer :

- la profondeur de passe suivant le rayon PR ,
- la profondeur de passe suivant la longueur PL ,
- le côté d'usinage SC , GA , DR ,

- un changement d'outil CO .

Si les profondeurs de passe ne sont pas programmées, elles valent zéro, ce qui fournit le profil fini.

2) Les informations relatives à un cycle d'usinage qui devront comporter :

- le numéro de section SE
- la longueur d'outil L
- la vitesse d'avance VA
- la vitesse de rotation $VR \begin{matrix} P \\ N \end{matrix}$
- la côte de déplacement rapide ZD
- la cote de perçage ZP
- le code du cycle voulu PEN , PEP, PEA, TA, ALA, ALT, ALR .

Exemple de phrases de données technologiques.

Exemple 1 : SE1 R10000 L 150000 VA 2300 VRP 1200
AR CO GA , PR 5000 , VRP 2000 PRO ;

Cette phrase précise dans l'ordre

- le numéro de section
- le rayon d'outil
- la longueur d'outil
- la vitesse d'avance
- la vitesse de rotation dans le sens positif
- une demande d'arrosage
- une demande de changement d'outil

- la position de la fraise à gauche du profil
- la profondeur de passe (suivant le rayon)

La virgule indique la fin du premier groupe d'informations.

- une nouvelle vitesse de rotation
- une nouvelle profondeur de passe qui correspond au profil fini. Il est indispensable de l'indiquer sinon on conserverait la valeur précédente.

Le point virgule indique la fin de la phrase.

Exemple 2 : SE1 L 150000 VA 1000 VRP 600 AR CO PEN ,
CO L 130000 VA 800 VRP 270 TA ;

Cette phrase d'informations relative à un perçage puis taraudage indique :

- la longueur d'outil
- la vitesse d'avance
- la vitesse de rotation dans le sens positif
- une demande d'arrosage
- une demande de changement d'outil
- une demande de cycle de perçage normal.

La virgule indique la fin du premier groupe d'informations.

- une demande de changement d'outil
- une nouvelle vitesse d'avance
- une nouvelle vitesse de rotation
- une demande de cycle de taraudage

Le point virgule indique la fin de la phrase.

Le nombre de virgule dans une phrase d'informations technologiques est illimité.

Remarque importante :

1) Avec le processeur GBS, il est toujours permis de mettre plusieurs groupes d'informations dans une même phrase (séparés par une virgule).

Par contre, avec GBM, il est interdit de le faire lorsque le fichier source est le clavier ou le lecteur du télétype. En effet, seul le lecteur rapide permet de dérouler la bande en arrière afin de pouvoir réexécuter la section avec chaque groupe d'informations technologiques.

2) Les longueurs d'outils doivent toutes être mesurées de la même manière et il faut que la côte zéro soit déterminée sur la machine à l'aide de l'outil utilisé dans la première section apparaissant dans le programme.

III-5-3 Modifications d'informations technologiques

Lorsque l'on dispose d'une bande symbolique définissant l'usinage d'une pièce avec certaines grandeurs technologiques et que celles-ci ne conviennent plus (outil modifié, vitesse d'avance mal adaptée, etc...) , cette bande permet de générer soit une nouvelle bande symbolique, soit une bande en format "machine", en utilisant le fichier MIT où se trouvent les nouvelles informations technologiques.

Sur ce fichier M T (modifications des informations technologiques) auquel peuvent être affectés, soit le clavier du télétype, soit le lecteur de celui-ci, doivent apparaître dans l'ordre, les numéros des sections dont les données technologiques sont à changer,

suivi de la phrase entière des nouvelles informations technologiques relative à la section appelée.

Lorsque le périphérique utilisé pour MIT est le clavier du télétype, on doit procéder comme suit.

Indiquer sur le télétype le numéro de la première section pour laquelle on désire modifier les grandeurs technologiques.

A ce moment là, le processeur GBS ou GBM lit la bande source et la traite jusqu'à la section annoncée sur le télétype. Quand le voyant "demande clavier" se rallume, on frappe les nouvelles informations technologiques. La section en cours est alors traitée en entier.

Quand le voyant "demande clavier" se rallume, on redonne un nouveau numéro de section et ainsi de suite.

Lorsque toutes les sections dont les informations technologiques devaient être changées, ont été traitées, on annonce sur le télétype un numéro de section inconstant sur la bande source afin que celle-ci soit traitée dans son intégralité.

III-5-4 Informations géométriques

Les définitions géométriques employées dans ce langage permettent de réaliser du contournage plan et du point à point.

Il est possible de distinguer 4 groupes de figures :

- les droites
- les cercles
- les structures de points linéaires
- les structures de points circulaires.

Pour les droites et les cercles, il existe un certain nombre de définitions possibles, dont certaines utilisent la figure précédente, la figure suivante ou les deux.

Pour les structures de points, les définitions sont plus limitées et ne font jamais appel aux blocs d'informations précédent ou suivant.

III-5-4-1 Généralités

Structure et particularités communes des blocs d'informations géométriques .

Une phrase d'informations géométriques est composée d'une succession de mots séparés par un espace au moins et dont l'ordre est indifférent, et se termine par un point virgule.

Nous allons tout d'abord examiner les différents types de définitions géométriques possibles.

Les mots précisant le mode de cotation, le côté d'usinage, la vitesse d'avance, etc... feront l'objet d'un chapitre ultérieur.

Les définitions géométriques utilisent très souvent les coordonnées X , Y , Z qui correspondent toujours au point final de la courbe définie dans le bloc, sauf dans le cas où ce point est déterminé par l'intersection avec la fonction suivante.

Les grandeurs EX et EY utilisées dans certaines définitions sont les coordonnées d'un point appartenant à la figure à définir, mais jamais le point initial ou le point final de celle-ci.

Toutes les grandeurs concernant des coordonnées ou des déplacements sont exprimées en microns à l'aide de 7 chiffres au maximum.

Les angles ou les déplacements angulaires sont exprimés en $1/10^{\text{ème}}$ de degré avec 4 chiffres maximum.

Définitions se référant aux figures suivantes ou précédentes.

Plusieurs types de mots permettent de définir des droites ou des cercles à partir de la figure précédente déjà définie, ou suivante précisée dans le bloc à venir.

- 1) TP Tangence à la figure précédente
- 2) TS Tangence à la figure suivante
- 3) TPS Tangence à la figure précédente et à la figure suivante
- 4) CP Intersection avec figure précédente
- 5) CS Intersection avec figure suivante

Ces tangences ou intersections comportant souvent plusieurs solutions, il faut donner des indications supplémentaires qui permettent de distinguer la solution désirée.

Ainsi, lorsque l'on veut définir le point d'intersection d'une droite avec le cercle précédemment programmé, on peut écrire par exemple l'instruction :

CP/PX , ce qui signifie qu'il s'agit du point d'intersection de la droite et du cercle qui a la plus petite composante suivant X. On peut ainsi programmer :

CP/PX : Point admettant la plus petite composante suivant X
CP/PY : Point admettant la plus petite composante suivant Y
CP/GX : Point admettant la plus grande composante suivant X
CP/GY : Point admettant la plus grande composante suivant Y

De même pour l'instruction CS.

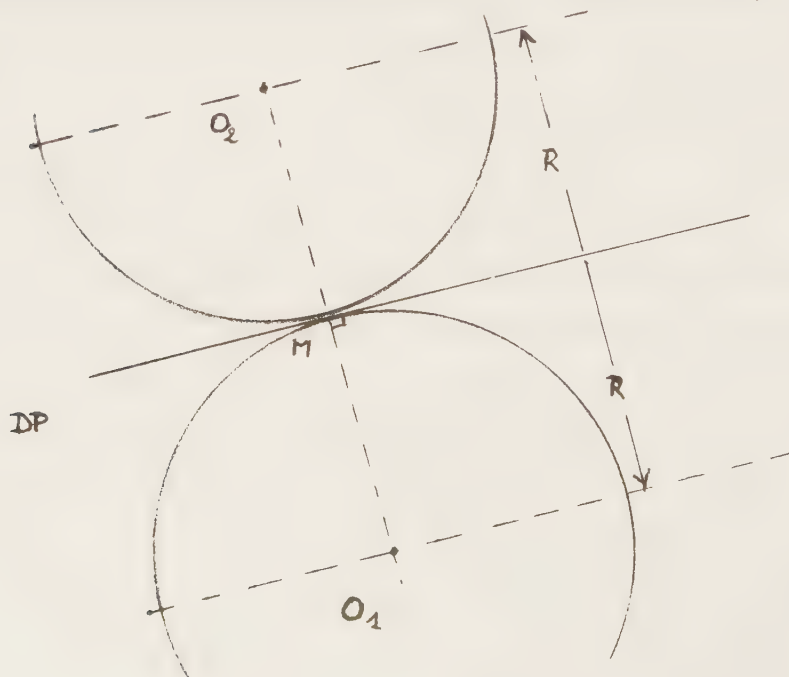
Pour les instructions de tangence, nous utiliserons deux façons de lever l'ambiguïté.

La première est identique à la définition énoncée pour les ordres CP et CS et sert dans tous les cas où l'inconnue sera le point de tangence.

La seconde permet de déterminer le centre d'un cercle tangent à la figure précédente ou suivante ou même aux deux.

Exemple 1 :

Soit à définir le cercle de rayon R tangent au point M à une droite DP connue (définie dans le bloc précédent).



On indique la position du centre par rapport à la droite DP, au-dessus ou au-dessous, respectivement par les signes + ou -

TP/+	ou	TP/-
pour O_2		pour O_1

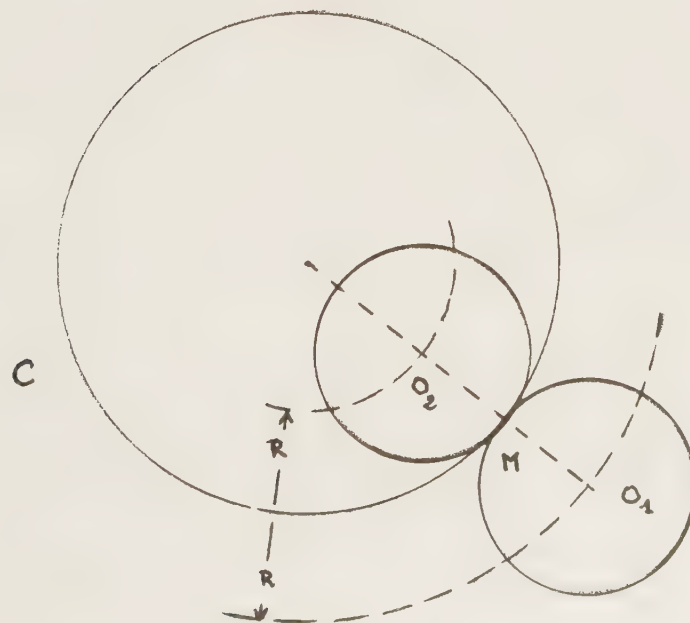
Lorsque la droite est verticale, on indique la position du centre du cercle tangent en précisant s'il est à droite (+) ou à gauche (-).

Exemple 2 :

On cherche le centre d'un cercle de rayon R tangent au cercle précédent C au point M.

On repère la solution choisie en précisant si les cercles sont tangents intérieurement (-) ou extérieurement(+).

Ainsi pour O_1 il faudra programmer TP/+ et pour O_2 TP/-



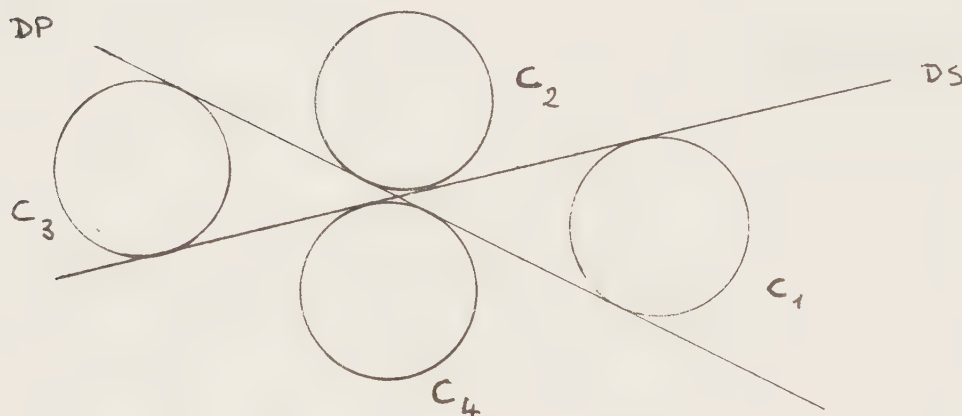
Pour l'instruction TPS, il faut donner 2 mots permettant de lever les indéterminations, pour la figure précédente et la figure suivante.

Exemple :

Soit à définir le cercle C tangent à la droite "précédente DP" et à la droite "suivante DS".

Le cercle C1 est repéré par

TPS/+/-



Dans les définitions que nous allons examiner, les différentes possibilités sont indiquées de la manière suivante :

Exemple : TPS/ $\begin{vmatrix} + \\ - \end{vmatrix} / \begin{vmatrix} + \\ - \end{vmatrix}$ ou CP/ $\begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix}$

D'autre part, les mots entre parenthèses sont des instructions qui ne doivent pas apparaître obligatoirement dans la définition, mais qui modifient celle-ci. Exemple : chapitre V-4-2-1 alinéa 1) .

Les instructions faisant appel au bloc précédent peuvent dans de nombreux cas être remplacées par la même instruction faisant appel au bloc suivant, mais programmée dans le bloc précédent.

Ainsi l'instruction CP programmée dans le bloc N 210 par exemple peut être remplacée par l'instruction CS programmée dans le bloc N 209.

En général, il est préférable d'utiliser le plus souvent possible des instructions faisant appel au bloc suivant, ceci afin que le tracé soit plus clair, en évitant de représenter des portions de figures superflues.

III-5-4-2 Les droites

Nous utilisons les définitions suivantes :

- droite passant par deux points
- droite passant par un point et ayant une pente donnée
- droite tangente à un cercle et passant par un point
- droite tangente à un cercle et ayant une pente donnée

Par ailleurs, la droite étant définie, les points initial et final peuvent être déterminés en considérant les intersections avec la figure précédente, la figure suivante ou les deux.

Nous séparons les définitions en trois groupes distincts :

- droites définies sans considération de tangence
- droites tangentes au cercle précédent
- droites tangentes au cercle suivant

Les blocs d'informations relatifs à une droite peuvent ou non

être précédés de la fonction $\$ \text{DRT} \begin{vmatrix} P \\ N \end{vmatrix}$. Le caractère P ou N indique le sens de parcours sur la droite avec les conventions suivantes :

- P : la coordonnée X croît du point initial au point final
- N : la coordonnée X décroît du point initial au point final.

Si les points final et initial ont la même coordonnée X , il faudra alors uniquement dans ce cas se référer à la coordonnée Y.

Dans les définitions suivantes, on indiquera la fonction $(\$ \text{DRT} \begin{vmatrix} P \\ N \end{vmatrix})$ entre parenthèses, s'il n'est pas nécessaire de la programmer.

Cette fonction n'est obligatoire que dans le cas où le point final n'est pas déterminé, d'une façon ou d'une autre, dans le même bloc d'informations.

III-5-4-2-1 Droites définies sans considération de tangence.

1) Droite passant par deux points :

$$\text{a) } (\$ \text{DRT} \begin{vmatrix} P \\ N \end{vmatrix}) \text{ X - Y - } (\begin{vmatrix} \text{CS} \\ \text{CS/} \end{vmatrix} \begin{vmatrix} \text{PX} \\ \text{PY} \\ \text{GX} \\ \text{GY} \end{vmatrix})$$

Il s'agit de la droite passant par le point précédemment défini et par le point programmé. Si ce point ne doit pas être le point final , on précise par le mot CS que celui-ci est le point d'intersection avec la figure suivante.

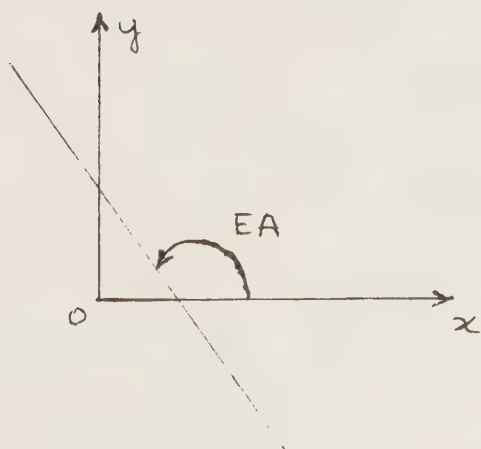
$$b) (\$ DRT \begin{vmatrix} P \\ N \end{vmatrix}) \text{ EX - EX - X - Y - } \begin{vmatrix} CP \\ CP / \end{vmatrix} \begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix}$$

Ce bloc définit la droite passant par les points de composantes EX , EY et X , Y . On doit alors définir le point initial en précisant qu'il s'agit du point d'intersection avec la figure précédente. Si celle-ci est une droite, l'ordre CP seul suffit puisqu'il n'existe alors qu'une solution, par contre pour un cercle il faut préciser que le point d'intersection convient.

2) Droite ayant une pente donnée et passant par un point :

Plusieurs possibilités se présentent :

$$a) \$ DRT \begin{vmatrix} P \\ N \end{vmatrix} \text{ EA - } (\begin{vmatrix} CS \\ CS / \end{vmatrix} \begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix})$$



Cette expression définit la droite passant par le point précédemment programmé et faisant avec le demi-axe ox un angle EA (en 1/10^{ème} de degré).

Si l'on ne précise pas qu'elle coupe la figure suivante, le bloc d'informations géométriques suivant devra comporter soit CP, soit TP, soit TPS, afin que le point final de cette droite soit déterminé.

$$b) (\$ DRT \begin{vmatrix} P \\ N \end{vmatrix}) X - Y - EA - \begin{vmatrix} CP \\ CP / \end{vmatrix} \begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix} \quad \left(\begin{vmatrix} CS \\ CS / \end{vmatrix} \begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix} \right) ;$$

Droite passant par le point de coordonnées X , Y et ayant une pente définie par EA.

Utilisée avec l'ordre CS, cette définition est identique à la précédente.

III-5-4-2-2 Droites tangentes au cercle précédent.

$$a) \$ DRT \begin{vmatrix} P \\ N \end{vmatrix} \quad TP \quad \left(\begin{vmatrix} CS \\ CS / \end{vmatrix} \begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix} \right) ;$$

Droite tangente au cercle précédent, au point final de l'arc considéré.

$$b) (\$ DRT \begin{vmatrix} P \\ N \end{vmatrix}) \quad TP / \begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix} \quad X - Y - \left(\begin{vmatrix} CS \\ CS / \end{vmatrix} \begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix} \right) ;$$

Droite tangente au cercle précédent et passant par le point de coordonnées X , Y .

$$c) (\$ DRT \begin{vmatrix} P \\ N \end{vmatrix}) \quad TP / \begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix} \quad EA - \left(\begin{vmatrix} CS \\ CS / \end{vmatrix} \begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix} \right) ;$$

Droite tangente au cercle précédent et admettant une pente définie par EA.

III-5-4-2-3 Droites tangentes au cercle suivant .

$$a) (\$ DRT \begin{vmatrix} P \\ N \end{vmatrix}) \quad TS / \begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix} ;$$

Droite passant par le point précédemment programmé et tangente au cercle suivant.

$$b) (\$ DRT \begin{vmatrix} P \\ N \end{vmatrix}) \quad TS / \begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix} \quad EX - EY - \begin{vmatrix} CP \\ CP \end{vmatrix} / \begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix} ;$$

Droite tangente au cercle suivant, passant par le point de coordonnées EX , EY .

$$c) (\$ DRT \begin{vmatrix} P \\ N \end{vmatrix}) \quad TS / \begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix} \quad EA - \begin{vmatrix} CP \\ CP \end{vmatrix} / \begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix} ;$$

Droite tangente au cercle suivant, ayant une pente définie par EA .

$$d) (\$ DRT \begin{vmatrix} P \\ N \end{vmatrix}) \quad TS \quad X - Y - \begin{vmatrix} CP \\ CP \end{vmatrix} / \begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix}$$

Droite tangente au cercle suivant, au point de coordonnées X , Y .

III-5-4-3 Les cercles

Un cercle peut être défini par :

- le centre et le rayon
- deux points de passage et le rayon
- un point de passage et le centre
- le centre et une tangence à un cercle ou une droite
- une tangence à un cercle ou une droite, un point de passage et le rayon
- une tangence à un cercle, le point de tangence et un point de passage
- une tangence à deux figures distinctes (cercles ou droites) et le rayon.

De même que pour les droites, le cercle étant défini, il est possible de déterminer les points initial et final en considérant les intersections avec les figures précédentes et suivantes.

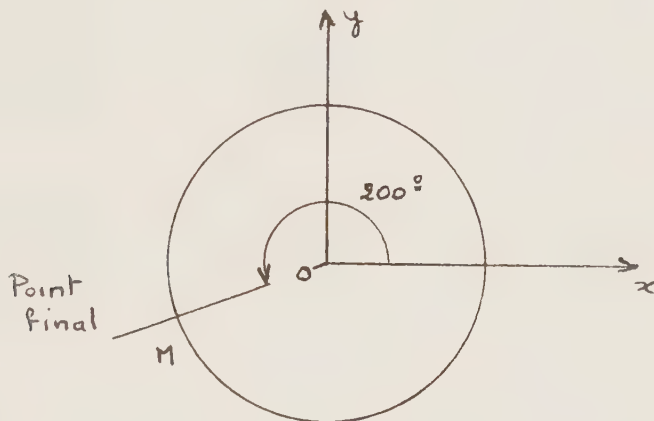
Quatre groupes distincts de cercles seront considérés :

- les cercles définis sans considération de tangence
- les cercles tangents à la figure précédente
- les cercles tangents à la figure suivante
- les cercles tangents aux figures précédente et suivante.

La définition d'un cercle sera toujours précédée de la fonction
 $\$ \text{CER} \begin{array}{|l} \text{P} \\ \text{N} \end{array}$. Le caractère P ou N indiquant si le cercle est à décrire dans le sens trigonométrique (P) ou rétrograde (N).

Le point final d'un cercle peut être repéré par l'angle (ox , OM). Cet angle est exprimé à l'aide du mot EA suivi d'un nombre

positif exprimé en 1/10ème de degré. Ainsi pour l'exemple ci-dessous le point final M pourra être déterminé par la fonction EA 2000.



III-5-4-3-1 Cercles définis sans considération de tangence

1) Cercle dont on connaît le centre et le rayon

$$\$ CER \begin{array}{c} P \\ N \end{array} \quad R - I - J - CP / \begin{array}{c} PX \\ PY \\ GX \\ GY \end{array} \quad (CS / \begin{array}{c} PX \\ PY \\ GX \\ GY \end{array}) ;$$

I et J sont respectivement l'abscisse et l'ordonnée du centre du cercle.

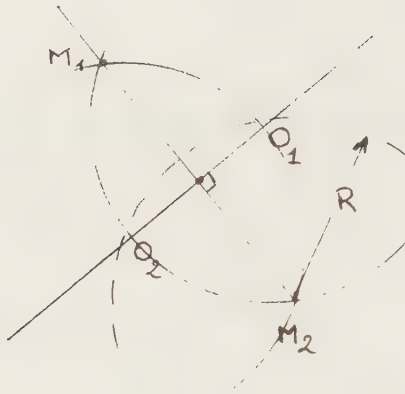
2) Cercle passant par deux points et dont on connaît le rayon.

Dans tous les cas, deux possibilités se présentent, comme le montre la figure ci-dessous :

- M_1 et M_2 sont les deux points de passage
- O_1 et O_2 sont les deux solutions possibles pour le centre du cercle.

Pour lever l'indétermination, il faudra préciser à l'aide de l'ordre

$$CP / \begin{array}{c} PX \\ PY \\ GX \\ GY \end{array} \quad \text{le centre qui convient.}$$



Ainsi, si O_1 est la solution cherchée, on doit faire apparaître dans la définition du cercle CP/GX ou CP/GY.

L'ordre CP utilisé ici est le même que l'ordre CP d'intersection avec la figure précédente mais est interprété d'une autre façon par le processeur. On ne doit donc pas définir un tel cercle, si l'on ne connaît pas le point initial autrement que par une intersection avec la figure précédente. Mais cela n'apporte aucune gêne puisqu'il est toujours possible, dans ce cas, de programmer une instruction CS (intersection avec figure suivante) dans le bloc d'information précédent.

$$a) \text{ § CER } \begin{array}{|c|} \hline P \\ \hline N \\ \hline \end{array} \quad X - Y - R - CP / \begin{array}{|c|} \hline PX \\ \hline PY \\ \hline GX \\ \hline GY \\ \hline \end{array} \quad (CS / \begin{array}{|c|} \hline PX \\ \hline PY \\ \hline GX \\ \hline GY \\ \hline \end{array}) ;$$

Cercle passant par le point précédemment programmé et par le point de coordonnées X , Y et de rayon R.

$$b) \text{ § CER } \begin{array}{|c|} \hline P \\ \hline N \\ \hline \end{array} \quad EX - EY - X - Y - R \quad CP / \begin{array}{|c|} \hline PX \\ \hline PY \\ \hline GX \\ \hline GY \\ \hline \end{array} \quad (CS / \begin{array}{|c|} \hline PX \\ \hline PY \\ \hline GX \\ \hline GY \\ \hline \end{array}) ;$$

Cercle passant par les points de coordonnées EX , EY et X , Y et admettant un rayon R.

Cette phrase ne pourra être utilisée qu'après un bloc où sera programmée une instruction CS.

3) Cercle passant par un point et dont on connaît le centre

$$a) \text{ § CER } \begin{vmatrix} P \\ N \end{vmatrix} \quad I - J - \quad (CS / \begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix}) ;$$

Cercle de centre I , J et passant par le point précédemment programmé.

$$b) \text{ § CER } \begin{vmatrix} P \\ N \end{vmatrix} \quad I - J - X - Y - CP / \begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix} (CS / \begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix}) ;$$

Cercle de centre I , J et passant par le point de coordonnées X , Y .

III-5-4-3-2 Cercles tangents à la figure précédente .

1) Cercle dont on connaît le point de tangence et le rayon

$$\text{ § CER } \begin{vmatrix} P \\ N \end{vmatrix} \quad TP / \begin{matrix} + \\ - \end{matrix} \quad R - (CS / \begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix}) ;$$

Cercle tangent à la figure précédente, au point précédemment programmé et de rayon R.

2) Cercle dont on connaît un point de passage et le rayon

$$\text{ § CER } \begin{vmatrix} P \\ N \end{vmatrix} \quad \begin{vmatrix} TP / \begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix} \\ TP / \begin{matrix} + \\ - \end{matrix} / \begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix} \end{vmatrix} \quad R - X - Y - (CS / \begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix}) ;$$

Cercle de rayon R tangent à la figure précédente et passant par le point de coordonnées X , Y .

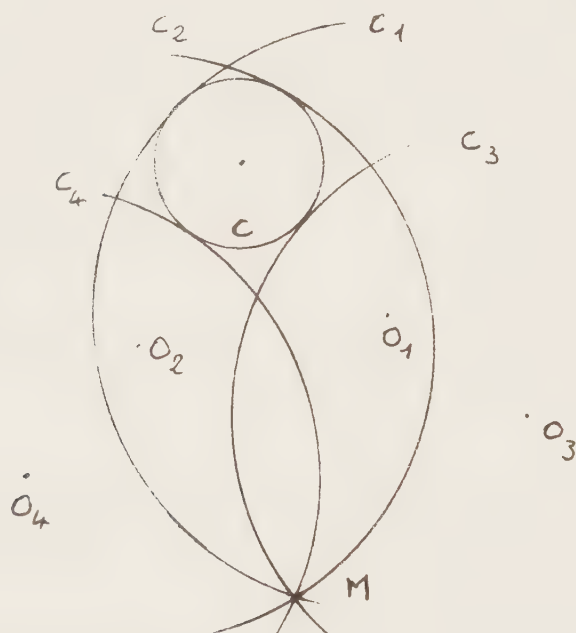
Si la figure précédente est une droite, il n'est pas nécessaire d'indiquer la position du centre par rapport à la droite, une seule solution étant possible.

Par contre, si la figure précédente est un cercle, on précise d'abord si les deux cercles sont tangents intérieurement (-) ou extérieurement (+) et ensuite quelle solution nous intéresse parmi les deux restantes.

Exemple.

C désigne le cercle précédent, M le point de passage. Quatre solutions C_1 , C_2 , C_3 , C_4 sont possibles. Si nous programmons par exemple TP / - / GX il s'agira du cercle C_1 de centre O_1 .

En effet C_1 est, parmi les cercles tangents intérieurement C_1 et C_2 de C, celui dont le centre a la plus grande abscisse.



3) Cercle dont on connaît le centre

$$\S \text{ CER } \left| \begin{array}{c} P \\ N \end{array} \right| \left| \begin{array}{c} TP \\ TP / \end{array} \right| \left| \begin{array}{c} PX \\ PY \\ GX \\ GY \end{array} \right| \quad I - J - (CS / \left| \begin{array}{c} PX \\ PY \\ GX \\ GY \end{array} \right|) ;$$

Si la figure précédente est une droite, il n'y a qu'un point de tangence. Sinon, il faudra préciser celle des deux solutions qui convient.

- 4) Cercle dont on connaît le point de tangence et un point de passage

$$\$ \text{CER} \begin{vmatrix} P \\ N \end{vmatrix} \quad \text{TP } X - Y - (CS / \begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix}) ;$$

Cercle qui est tangent à la figure précédente au point précédemment programmé et qui passe par le point de coordonnées X , Y . Il n'y a qu'une solution possible.

III-5-4-3-3 Cercles tangents à la figure suivante .

- 1) Cercle dont on connaît le point de tangence et le rayon

$$\$ \text{CER} \begin{vmatrix} P \\ N \end{vmatrix} \quad \text{TS} / \begin{matrix} + \\ - \end{matrix} \quad R - X - Y - CP / \begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix} ;$$

Cercle de rayon R tangent à la figure suivante au point de coordonnées X , Y .

- 2) Cercle dont on connaît le point de tangence et un point de passage

$$\$ \text{CER} \begin{vmatrix} P \\ N \end{vmatrix} \quad \text{TS} \quad X - Y - ;$$

Il s'agit du cercle passant par le point précédemment programmé et tangent à la figure suivante au point de coordonnées X , Y .

- 3) Cercle dont on connaît un point de passage et le rayon

$$\$ \text{CER} \begin{vmatrix} P \\ N \end{vmatrix} \quad \begin{vmatrix} \text{TS} / \\ \text{TS} / \end{vmatrix} \begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix} \begin{matrix} + \\ - \end{matrix} / \begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix} \quad R \text{ ——— } ;$$

Cercle de rayon R tangent à la figure suivante et passant par le point précédemment programmé.

Pour le lever d'indétermination sur le centre, on se référera au chapitre III-5-4-3-2 , alinéa 2.

4) Cercle dont on connaît le centre

$$\S \text{ CER } \begin{vmatrix} P \\ N \end{vmatrix} \quad \begin{vmatrix} TS \\ TS \end{vmatrix} / \quad \begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix} \quad I - J \quad CP / \quad \begin{vmatrix} PX \\ PY \\ GX \\ GY \end{vmatrix} ;$$

Cercle de centre dont les coordonnées sont I et J et tangent à la figure suivante.

On précise le point de concours avec la figure précédente par l'instruction CP.

III-5-4-4 Cercles tangents aux figures précédentes et suivantes

Seule une définition a été permise ici :

$$\S \text{ CER } \begin{vmatrix} P \\ N \end{vmatrix} \quad \text{TPS} / \quad \begin{vmatrix} + \\ - \end{vmatrix} / \quad \begin{vmatrix} + \\ - \end{vmatrix} \quad R \text{ ———— } ;$$

Il s'agit du cercle tangent à la figure précédente et à la figure suivante, admettant un rayon R. Parmi toutes les possibilités offertes, celle qui convient est déterminée en précisant, après l'instruction TPS, la position du centre par rapport aux figures précédente et suivante.

III-5-4-5 Les structures de point

Des structures de points linéaires et circulaires peuvent être définies. La droite ou le cercle sur lequel sont disposés les points ne pourra pas être défini à partir de la figure précédente ou suivante.

Un bloc d'informations définissant une structure de points doit toujours comporter le point initial, l'écart angulaire ou la distance entre les points et le nombre de ceux-ci.

III-5-4-5-1 Structure linéaire .

Deux définitions sont possibles :

$$1) \text{ § PDR } \left| \begin{array}{c} P \\ N \end{array} \right. \text{ EX - EY - DX - DY - NP - } ;$$

EX , EY sont les coordonnées du point initial

DX , DY sont les composantes du vecteur qui fait passer d'un point au suivant.

NP est le nombre de points de la structure

Il faut remarquer que dans cette définition le caractère P ou N qui suit la fonction § PDR et qui précise le sens de parcours sur la droite est une information redondante. En effet, ce sens de parcours est défini, sans équivoque, par DX et DY. Nous avons, malgré tout, gardé cette forme de définition par souci d'homogénéité avec les autres fonctions.

$$2) \text{ § PDR } \left| \begin{array}{c} P \\ N \end{array} \right. \text{ EX - EY - EA - DL - NP — } ;$$

EX , EY sont les coordonnées du point initial

EA est l'angle formé par la droite et le demi-axe Ox (toujours positif).

DL est la distance entre deux points successifs

NP est le nombre de points de la structure

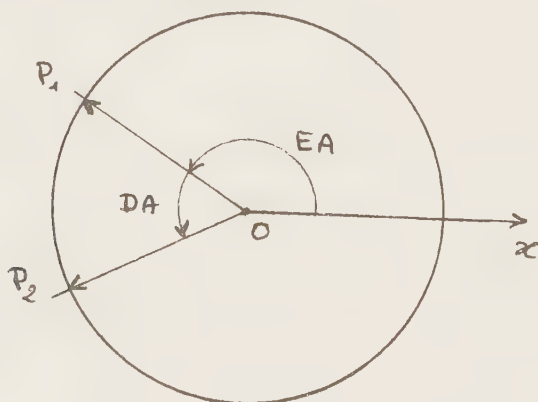
Dans cette définition, le caractère P ou N revêt une importance toute particulière car c'est lui seul qui précise le sens de succession des différents points.

III-5-4-5-2 Structure circulaire.

Deux définitions sont possibles pour définir un ensemble de points répartis sur un cercle :

$$1) \text{ § PCE } \left| \begin{array}{l} \text{P} \\ \text{N} \end{array} \right| \text{ I - J - R - EA - DA - NP - ;}$$

Cette instruction décrit un ensemble de points répartis sur le cercle de rayon R et dont le centre O admet pour coordonnées I et J. Le premier point P_1 de la structure est déterminé par l'angle EA (voir figure).



Les points successifs sont déterminés par DA qui est l'écart angulaire entre deux points voisins.

NP exprime le nombre de points de la structure.

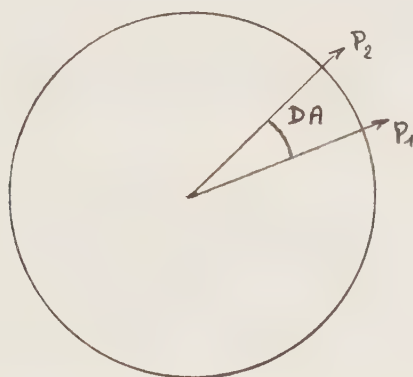
Remarque :

DA est un angle positif, le sens de succession des points sur le cercle est déterminé par le caractère P ou N qui succède à la fonction § PCE .

$$2) \text{ § PCE } \begin{array}{l} \text{P} \\ \text{N} \end{array} \text{ I - J - EX - EY - DA - NP - ;}$$

Le cercle sur lequel sont répartis les points est déterminé sans équivoque par son centre O de coordonnées I et J et le premier point P_1 de coordonnées EX - EY .

DA et NP expriment les mêmes grandeurs que dans la définition précédente.

Remarques concernant les fonctions faisant appel au bloc suivant

Lorsque l'on programme une telle fonction, le calculateur attend le prochain bloc d'informations et exécute le tracé de la figure (si celui-ci est demandé) après le deuxième point virgule. Si le tracé est correct, l'ordre de validation agit sur les deux blocs.

Par contre, si une erreur est détectée après le deuxième point virgule, ou si le tracé ne correspond pas à ce qui est attendu, il est nécessaire de refrapper les deux blocs d'informations dans leur totalité.

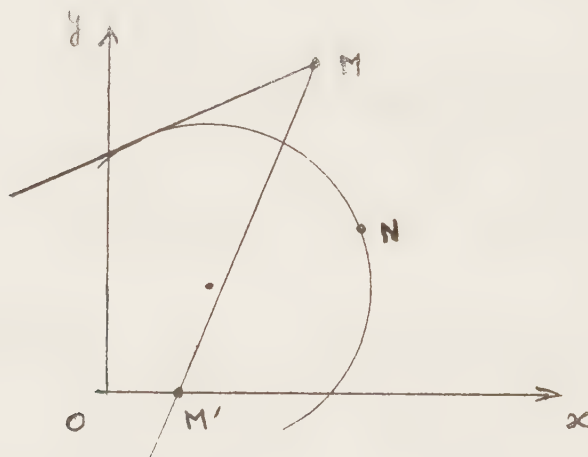
Lorsqu'une fonction nécessite la connaissance du bloc suivant, celui-ci devra, bien entendu, définir une figure sans faire appel au bloc suivant. Mais il est possible que cette figure passe par un point précédemment programmé. Dans ce cas, ce point sera non pas celui qui pourrait être programmé dans le dernier bloc mais celui qui apparaît dans l'avant dernier. Examinons l'exemple suivant :

```

$ DRTP  X 1 00000 Y 2 00000 ;
$ CERN  TP / PX R 1 00000 X 8 0000 Y 1 00000 CS / PY ;
$ DRTN  X 2 0000 Y 0 ;

```

La droite définie dans ce dernier bloc est la droite passant par le point M de coordonnées $\begin{vmatrix} 100\ 000 \\ 200\ 000 \end{vmatrix}$ et par le point M' de coordonnées $\begin{vmatrix} 20\ 000 \\ 0 \end{vmatrix}$ et non pas par le point N de coordonnées $\begin{vmatrix} 80\ 000 \\ 100\ 000 \end{vmatrix}$



III-5-4-6 Commandes et ordres divers

Les blocs d'informations géométriques peuvent comporter des mots de commande apportant des précisions sur le mode de cotation, le côté d'usinage, etc...

1) Mode de cotation : il existe quatre ordres précisant le mode de cotation :

- AB : cotation absolue (pour le bloc où cet ordre apparaît seulement)
- ABR : cotation absolue rémanente
- RE : cotation relative (pour le bloc où cet ordre apparaît seulement)
- RER : cotation relative rémanente

Au départ, le mode de cotation est initialisé automatiquement en absolu rémanent. Les modes non rémanents permettent de changer, pour un bloc d'information seulement, le mode de cotation dans lequel la pièce est programmée. Il faut noter que les ordres ci-dessus peuvent apparaître n'importe où dans le bloc et même être répétés, seul le dernier étant pris en compte.

En mode de cotation relative, toutes les grandeurs définissant des coordonnées sont cotées en relatif par rapport au dernier point programmé et dont les coordonnées ont été définies par X , Y ou Z.

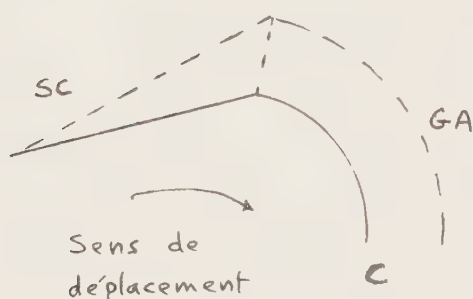
2) Définition du côté d'usinage

Pour préciser de quel côté se trouve la fraise par rapport au profil fini, trois mots sont utilisés :

- SC : sans correction, déplacement du centre de la fraise sur le profil programmé ,
- DR : droite. Déplacement de l'outil à droite du profil dans le sens du déplacement,
- GA : gauche. Déplacement de l'outil à gauche du profil.

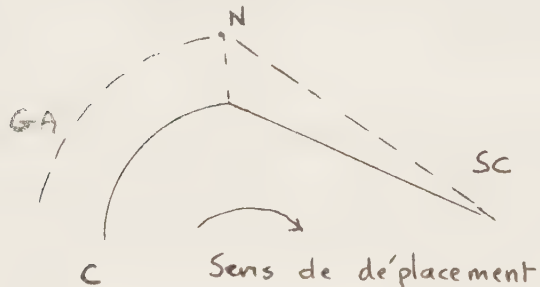
Remarques importantes

Il est interdit de passer de gauche à droite d'un profil sans programmer un bloc où apparaisse la commande SC (sans correction). D'autre part, cette commande ne devra pas apparaître dans un bloc définissant un cercle. Le passage d'un bloc sans correction à un bloc avec correction se fera de la manière suivante :



La figure programmée est en trait plein, le centre de la fraise se déplaçant selon le trait interrompu. La droite N est la normale au cercle C.

Le passage inverse sera réalisé de la façon suivante :



En procédant de cette manière, l'outil ne mordra à aucun moment dans le profil pour lequel on a programmé une correction (ici le cercle C)

On notera également qu'au début d'une section, quel que soit le côté d'usinage programmé, le premier bloc sera considéré sans correction et devra donc être une droite.

3) Commandes relatives à la vitesse d'avance

Trois modes de vitesse sont permis. Les ordres correspondants sont :

VTR : Vitesse travail rémanente

VR : Vitesse rapide (pour le bloc où cet ordre apparaît seulement)

VRR : Vitesse rapide rémanente

Au début du programme , le mode de vitesse d'avance est initialisé automatiquement en VTR (vitesse travail rémanente).

L'ordre VR permet l'usinage en vitesse rapide du seul bloc où il apparaît.

4) Ordres FS et FIN

Le mot FS indique une fin de section. Il doit apparaître seul suivi d'un point virgule. Après cet ordre devra suivre une phrase d'informations technologiques ou le mot FIN qui indique la fin du programme. Il doit également apparaître seul suivi d'un point virgule et être toujours précédé de l'ordre FS, sinon la dernière section programmée ne sera pas totalement prise en compte si la phrase d'informations technologiques correspondante est formée de plusieurs groupes de données.

5) Le caractère

Ce caractère sert de validation des blocs d'informations géométriques.

Après le point virgule, fin d'un bloc d'informations, la fonction programmée est tracée sur la table traçante (si toutefois cela a été demandé). Si ce tracé convient il suffira de frapper le caractère % pour que le bloc en question soit pris en compte.

Sinon, il faudra taper un caractère quelconque, par exemple un retour chariot, ce qui aura pour effet d'annuler le bloc et de ramener la plume de la table traçante à la fin de la figure précédemment décrite.

Il ne faut surtout pas oublier le caractère % lorsque le programme symbolique est frappé sur le télétype off-line. Dans ce cas, on veillera à ne pas taper ce caractère après un bloc définissant une fonction faisant appel au prochain bloc d'informations.

D'autre part, si on n'a pas demandé de tracé de la pièce programmée, il ne faut pas pour autant omettre les caractères de validations, sinon aucun bloc ne serait pris en compte.

III-5-5 Lancement des processeurs GBS et GBM

Lorsqu'ils ont été appelés les processeurs GBS et GBM attendent des informations issues du clavier (voyant demande clavier allumé).

Ces informations vont permettre de définir l'échelle et la vue, si nécessaire, du tracé de la pièce.

On procédera comme suit :

- Si la première phrase d'informations technologiques qui apparaît dans le programme définit également les grandeurs relatives au tracé sur table traçante, il ne sera pas nécessaire de les préciser.

On indiquera alors seulement si l'on désire faire apparaître les déplacements en vitesse rapide. Si oui, on frappera la commande "VR ; ", sinon il suffira de taper le caractère point virgule pour lancer le programme.

- Si la première phrase d'informations technologiques ne comporte pas la définition des grandeurs relatives au tracé de la pièce, il faut alors les faire apparaître avant de lancer le programme.

Les mots permettant de préciser l'échelle et la vue désirée sont identiques à ceux qui peuvent apparaître dans une phrase de données technologiques (voir V-2). Mais il est possible ici de préciser si l'on désire ou non le tracé des déplacements en vitesse rapide. Dans ce dernier cas, on fera apparaître le mot "VR" dans cette phrase qui se terminera toujours par un point virgule.

Si l'on ne désire aucun tracé de la pièce, il suffira de frapper un point virgule seul.

III-6 Utilisation du système d'aide à la programmation :

La marche normale à suivre correspondant à la meilleure utilisation du système d'aide à la programmation proposé est la suivante :

1) Etablissement du programme symbolique de la pièce à usiner ;

2) Frappe de ce programme en conversationnel avec le processeur GBS (génération de bandes symboliques). Tracé sur table traçante de chaque bloc et validation de ceux qui conviennent;

3) Génération d'une bande "machine" à l'aide du processeur GBM, à partir de la bande objet obtenue dans l'étape précédente. Tracé complet de la pièce si nécessaire.

Les deux bandes obtenues dans les étapes 2 et 3 sont à conserver.

La bande établie par GBS sera très utile lorsque certaines grandeurs technologiques devront être changées ; en effet, il suffit alors d'utiliser le fichier MIT (voir chapitre V-3).

On peut utiliser ce système d'aide à la programmation de bien d'autres manières. En particulier, le déroulement suivant peut présenter certains avantages :

- 1) Etablissement du programme symbolique de la pièce à usiner ,
- 2) Frappe de ce programme off line ,
- 3) Génération d'une bande "machine" à partir de la bande obtenue "off line" , avec détection des erreurs. Tracé de la pièce si nécessaire.

Dans ce mode d'utilisation, le calculateur est occupé moins longtemps, mais on perd tout l'avantage de dialogue et de correction immédiate possibles par les processeurs.

D'autre part, si l'on désire conserver une bande dont on puisse modifier les données technologiques, il faut corriger avec COB les éventuelles erreurs de la bande source obtenue "off line".

EXEMPLE DE PROGRAMMATION

* * *

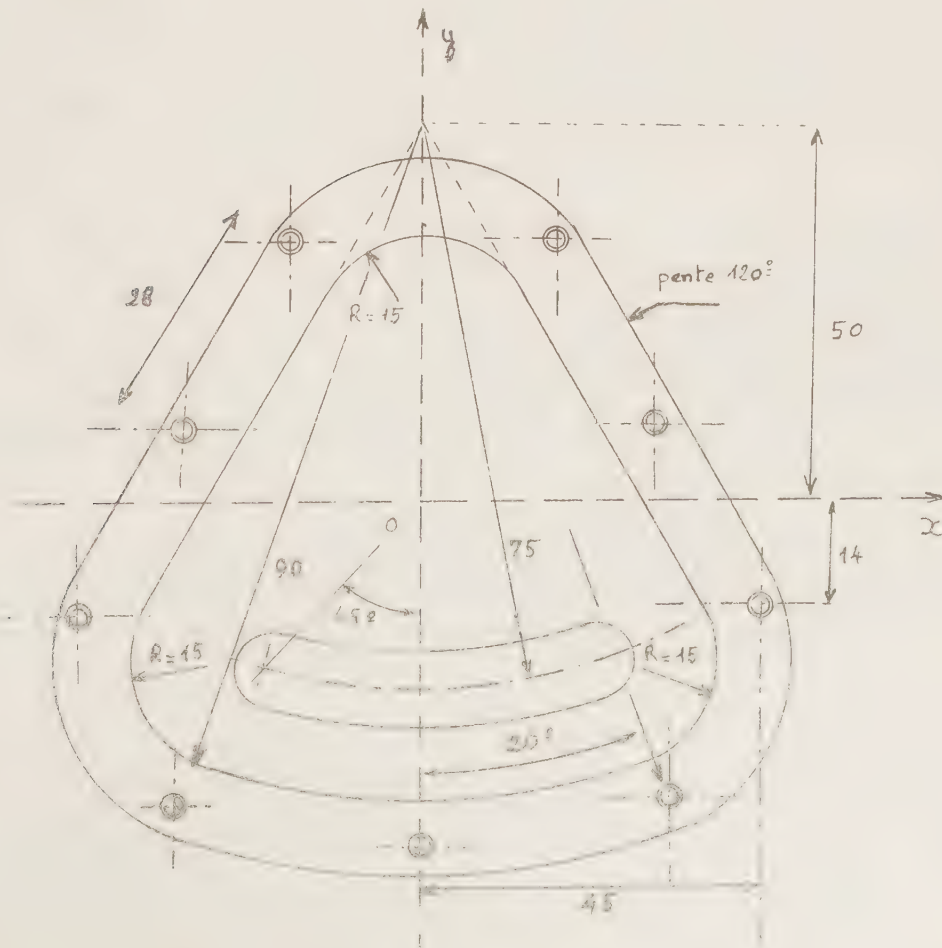
Nous avons programmé la pièce ci-dessous en deux passes.

La première avec GBS dont le résultat est une bande symbolique où les fonctions apparaissent sous forme canonique.

La seconde avec GBM qui délivre la bande destinée à l'armoire de commande de la machine outil.

Nous donnons dans les pages suivantes un aperçu de la programmation en conversationnel de cette pièce sur le téléimprimeur (les phrases issues du calculateur sont soulignées), ainsi que le tracé obtenu sur la table traçante au cours de cette opération.

On trouvera ensuite le listing de la bande symbolique objet.



SE1 CO DR L100000 R10000 VA1200 VRP1000 PR25000 ,
 SFO1 NAR CO DR R0010000 PR0025000 L0100000 PL0000000 VA1200 VRP1000 ,
 PR5000
 , NAR DR R0010000 PR0005000 L0100000 PL0000000 VA1200 VRP1000 ,
 PPO
 , NAR DR R0010000 PR0000000 L0100000 PL0000000 VA1200 VRP1000 ,
 GA PR-5000 PL-3000
 , NAR GA R0010000 PR-0005000 L0100000 PL-0003000 VA1200 VRP1000 ;

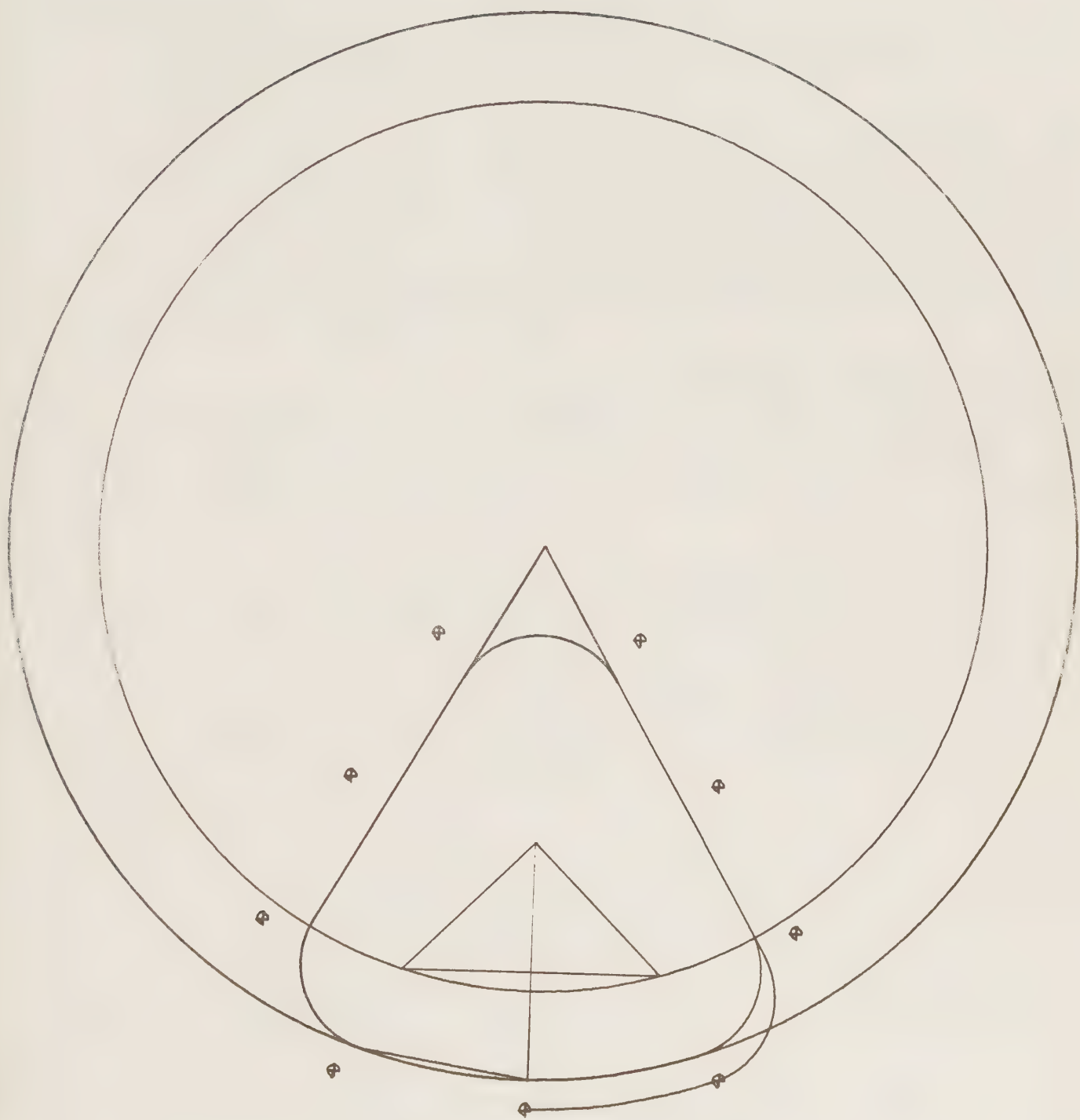
\$DRTN EA900 Z-50000 CS/PY ;
 \$CERN IO J50000 R90000 ; %
 NO01 X-0000000 Y-0040000 Z-0050000 ; %
 \$CERN TPS/-/-/PY ;
 \$DRTN XO Y50000 EA600 ;
 MANO INF FONC
 \$CERN R15000 TPS/-/-/PY ;
 \$DRTN XO Y50000 EA600 ; %
 NO02 \$CERN X-0028504 Y-0035372 I0000000 J0050000 F0090000 ; %
 NO03 \$CERN X-0036714 Y-0013590 I-0023752 J-0021139 R0015000 ; %
 \$CERN R15000 TPS/-/- ;
 \$DRTN EA1200 X50000 Y0 XO Y50000 ; %
 NO04 X-0012962 Y0027548 ; %
 NO05 \$CERN X0012966 Y0027541 I-0000000 J0020000 R0015000 ; %
 \$CERN TPS/-/-/PY R15000 ;
 \$CERN IO J50000 EA2700 R95000 ;
 \$CERN TPS/-/-/PY R15000 ;
 MOT N CONF R15000 ;
 \$CERN IO J50000 R90000 ;
 \$CERN TPS/-/-/PY R15000 ;
 \$CERN IO J50000 R90000 EA2700 ; %
 NO06 X0036717 Y-0013597 ; %
 NO07 \$CERN X0028503 Y-0035373 I0023750 J-0021139 R0015000 ; %
 FS;
 NO08 \$CERN X-0000001 Y-0039999 I0000000 J0050000 R0090000 ; %FS;

SE2 CO L100000 R5000 VA1000 VRP1200 SC ;
 SE02 NAR CO SC R0005000 PR0000000 L0100000 PL0000000 VA1000 VRP1200 ;

VR XO YO ; %
 \$DRTN EA450 Z-55000 CSP
 MOT N CONF CS/PY ;
 \$CERN R75000 IO J50000 ZO ; %
 NO01 VR X0000000 Y0000000 ; %
 NO02 X-0021770 Y-0021770 Z-0055000 ; %
 \$DRTN XO YO EA1350 CP/GX ; %
 NO03 \$CERN X0021770 Y-0021770 Z0000000 I0000000 J0050000 R0075000 ; %
 FS;
 NO04 X0000000 Y0000000 ; %FS;

SE3 CO L100000 VA1200 VRP1200 ZD-45000 ZP-57000 PEN,
 SE03 NAR CO SC PEN ZD-0045000 ZP-0057000 L0100000 PL0000000 VA1200 VRP1200
 CO L100000 VA1000 VRP1000 TA
 ; NAR CO SC TA ZD-0045000 ZP-0057000 L0010000 PL0000000 VA1000 VRP1000 ;

\$PCEP EA2500 DA200 NP3 R95000 IO J50000 ; %
 \$PDRP EX45000 EY-14000 EA1200 DL23000 NP3 ; %
 NO01 \$PCEP I0000000 J0050000 DA0200 R0095000 NP003 ; %
 \$PDRP EX-45000 EY-14000 EA600 DL23000 NP3 ; %
 NO02 \$PDRP EX0045000 EY-0014000 DX-0013999 DY0024248 NP003 ; %
 FS;
 NO03 \$PDRP EX-0045000 EY-0014000 DX0014000 DY0024248 NP003 ; %FS;



SE01 NAR CO DP R0010000 PR0025000 L0100000 PL0000000 VA1200 VRP1000 ,
 NAR DR R0010000 PR0005000 L0100000 PL0000000 VA1200 VRP1000 ,
 NAR DR R0010000 PR0000000 L0100000 PL0000000 VA1200 VRP1000 ,
 NAR GA R0010000 PR-0005000 L0100000 PL-0003000 VA1200 VRP1000 ;

N001 X-0000000 Y-0040000 Z-0050000 ;%
 N002 SCERN X-0028504 Y-0035372 I0000000 J0050000 R0090000 ;%
 N003 SCERN X-0036714 Y-0013590 I-0023752 J-0021139 R0015000 ;%
 N004 X-0012962 Y0027548 ;%
 N005 SCERN X0012966 Y0027541 I-0000000 J0020000 R0015000 ;%
 N006 X0036717 Y-0013597 ;%
 N007 SCERN X0028503 Y-0035373 I0023750 J-0021139 R0015000 ;%
 N008 SCERN X-0000001 Y-0039999 I0000000 J0050000 R0090000 ;%FS;
 SE02 NAR CO SC R0005000 PR0000000 L0100000 PL0000000 VA1000 VRP1200 ;

N001 VR X0000000 Y0000000 ;%
 N002 X-0021770 Y-0021770 Z-0055000 ;%
 N003 SCERP X0021770 Y-0021770 Z0000000 I0000000 J0050000 R0075000 ;%
 N004 X0000000 Y0000000 ;%FS;
 SE03 NAR CO SC PEN ZD-0045000 ZP-0057000 L0010000 PL0000000 VA1200 VRP1000
 NAR CO SC TA ZD-0045000 ZP-0057000 L0010000 PL0000000 VA1000 VRP1000 ,
 NAR CO

N001 SPCEP I0000000 J0050000 DA0200 R0095000 MF003 ;%
 N002 SPIRP FX0045000 EY=0014000 DX=0013999 DY0024248 NP003 ;%
 N003 SPDRP FX-0045000 EY-0014000 DX0014000 DY0024248 NP003 ;%FS;

* * * * *

* *

* CONCLUSION *

* *

* * * * *

Le système d'aide à la programmation que nous avons étudié présente, par rapport aux langages existants, des avantages certains.

Le premier et le plus important de ceux-ci est le contrôle en conversationnel de la géométrie de la pièce par tracé direct des courbes correspondant aux instructions géométriques programmées.

Une deuxième qualité importante est la programmation simultanée des ordres d'usinage et des informations géométriques, celles-ci apparaissant strictement dans l'ordre où elles seront usinées.

D'autre part, le partage de la définition d'une pièce en plusieurs sections correspondant à des grandeurs technologiques différentes, assure à ce langage une grande souplesse dans la programmation des passes d'ébauches, des cycles multiples (perçage, taraudage) et surtout permet de modifier très facilement les paramètres de l'usinage.

Enfin ce système apporte une véritable aide à la programmation grâce aux processeurs qui permettent la correction des bandes perforées, la visualisation du profil fini de la pièce programmée et le tracé du trajet du centre de l'outil.

De plus, ne nécessitant aucune mémoire de masse, ce système d'aide à la programmation pourra s'intégrer facilement dans un contexte C. N. C. et devrait permettre à ce nouveau type de commande numérique d'être diffusé largement dans des entreprises de petite taille n'utilisant qu'un nombre limité de machines pour usiner des pièces de complexité moyenne.

 * BIBLIOGRAPHIE *

- [1] **Programmation Automatique des Machines Outils à
 Commande Numérique.**
 H. OPITZ
 B. HIRSCH
 Conférence européenne de Commande Numérique des
 Machines Outils
 CIRP 24 et 25 avril 1968
- [2] **La commande symbolique**
 Le système APT de programmation des Machines Outils
 à Commande Numérique.
 Benjamin MITTMAN
 Automatisme N° 2 Février 1965
- [3] **Langages de programmation en commande numérique**
 M. CHALVET
- [4] **Programme de Commande Numérique sur Calculateur**
 Hewlette Pakard modèle 10 . ELAN
 Manuel d'utilisation (CECN Lyon)
- [5] **KPROPS Notice d'utilisation**
 Bibliothèque RAX W16 - 0186
- [6] **MICROCAP** Olivetti Manuel d'utilisation

- [7] Utilisation des machines comptables pour la programmation
des machines outils à commande numérique

Rev ue C. N. N° 15 Juin-Juillet 1971

- [8] Un moyen de programmation économique et simple.

Revue C. N. N° 18 Décembre 1971

- [9] Le système "PROFILE-DATA"

J. Ponte - Mécanique - Août-Septembre 1968

- [10] ROMANCE

Bibliothèque IBM N° 1130 - 23 - 4 - 002 (Classe III)

- [11] Les langages de programmation automatique

.Conférence de Monsieur Gendre (CII)

ADECNA 18 Janvier 1973 - BORDEAUX.

* * *

 * ANNEXES *

TABLE DES MATIERES

I	Le superviseur	1
II	Programme TRM	7
III	Programme COB	1 2
IV	Processeurs TRS , GBS , GBM	1 5
V	Sous-programmes d'interpolation et de calculs géométriques	4 1
VI	Programmes de traitement des fonctions lues	6 8
VII	Sous-programmes de sortie des informations symboliques ou machines	7 6
VIII	Sous-programmes divers	8 5

I - Le Superviseur SUPMO.

Le superviseur permet l'appel du processeur que l'on désire mettre en oeuvre, et la sélection des périphériques à utiliser.

Les trois premiers caractères lus sont décodés et préparent le branchement SUP1 vers les programmes d'initialisation et d'appel des divers processeurs.

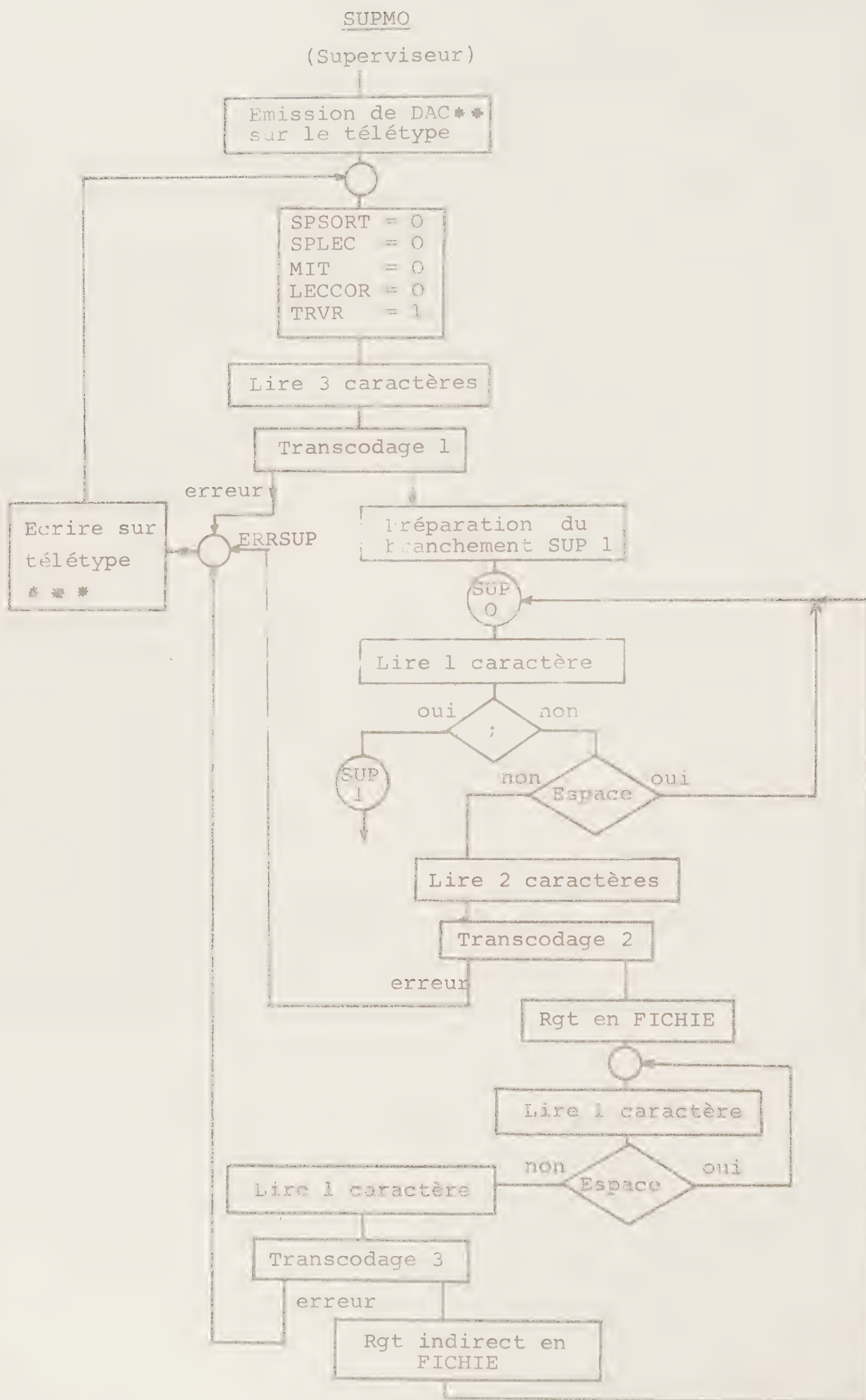
Les trois caractères suivants indiquent le fichier auquel on va attribuer un périphérique. Ils sont transcodés et l'on range en FICHIE l'adresse d'appel du sous-programme d'entrée-sortie relatif au fichier considéré.

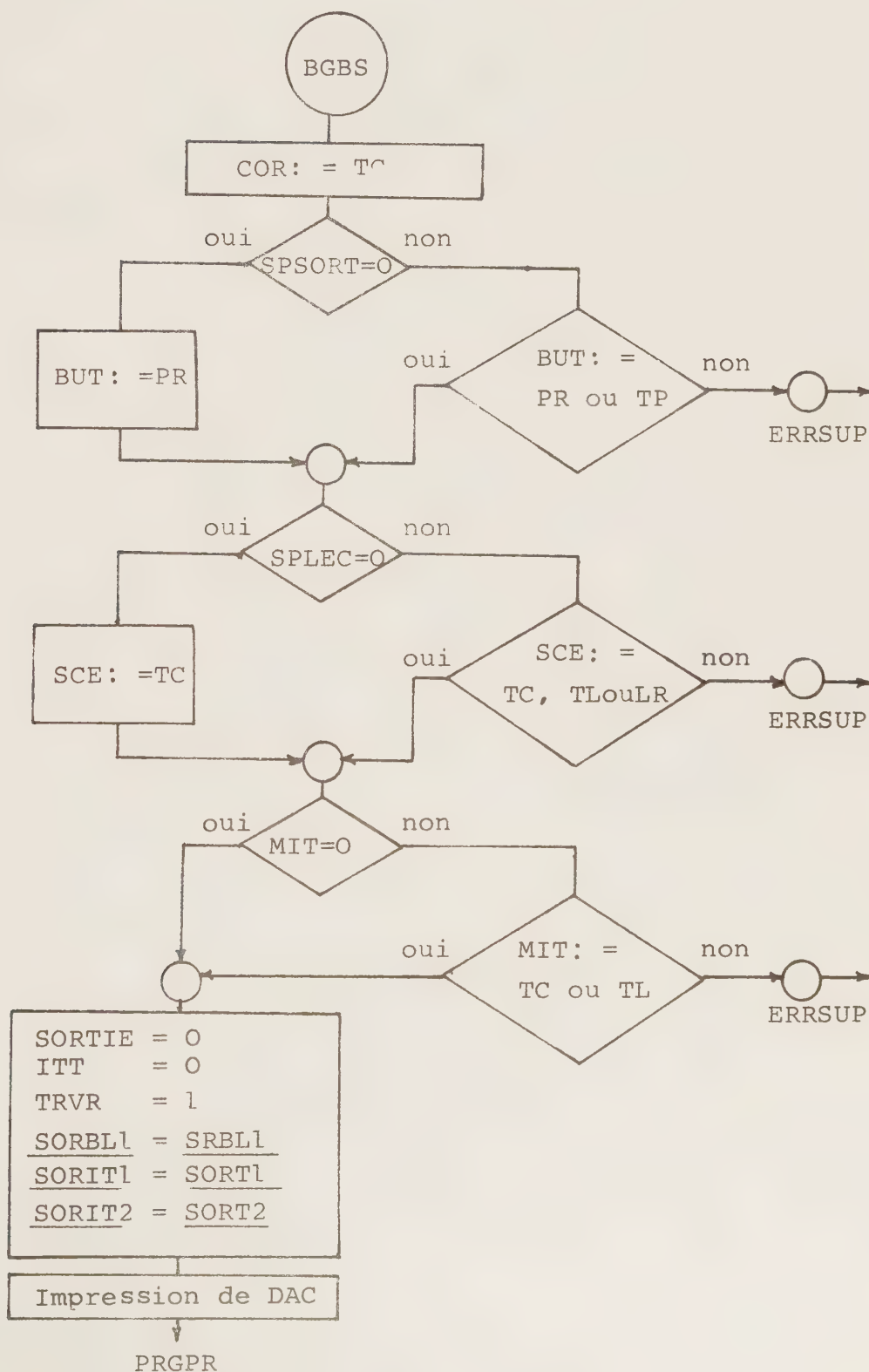
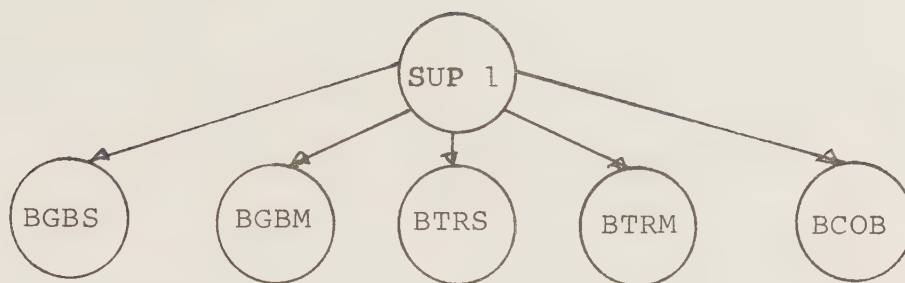
Les trois derniers caractères indiquent le périphérique attribué au fichier précédemment mentionné, ils sont transcodés et le résultat de cette opération, qui est une adresse de départ d'un sous-programme d'entrée-sortie, est rangé indirectement en FICHIE.

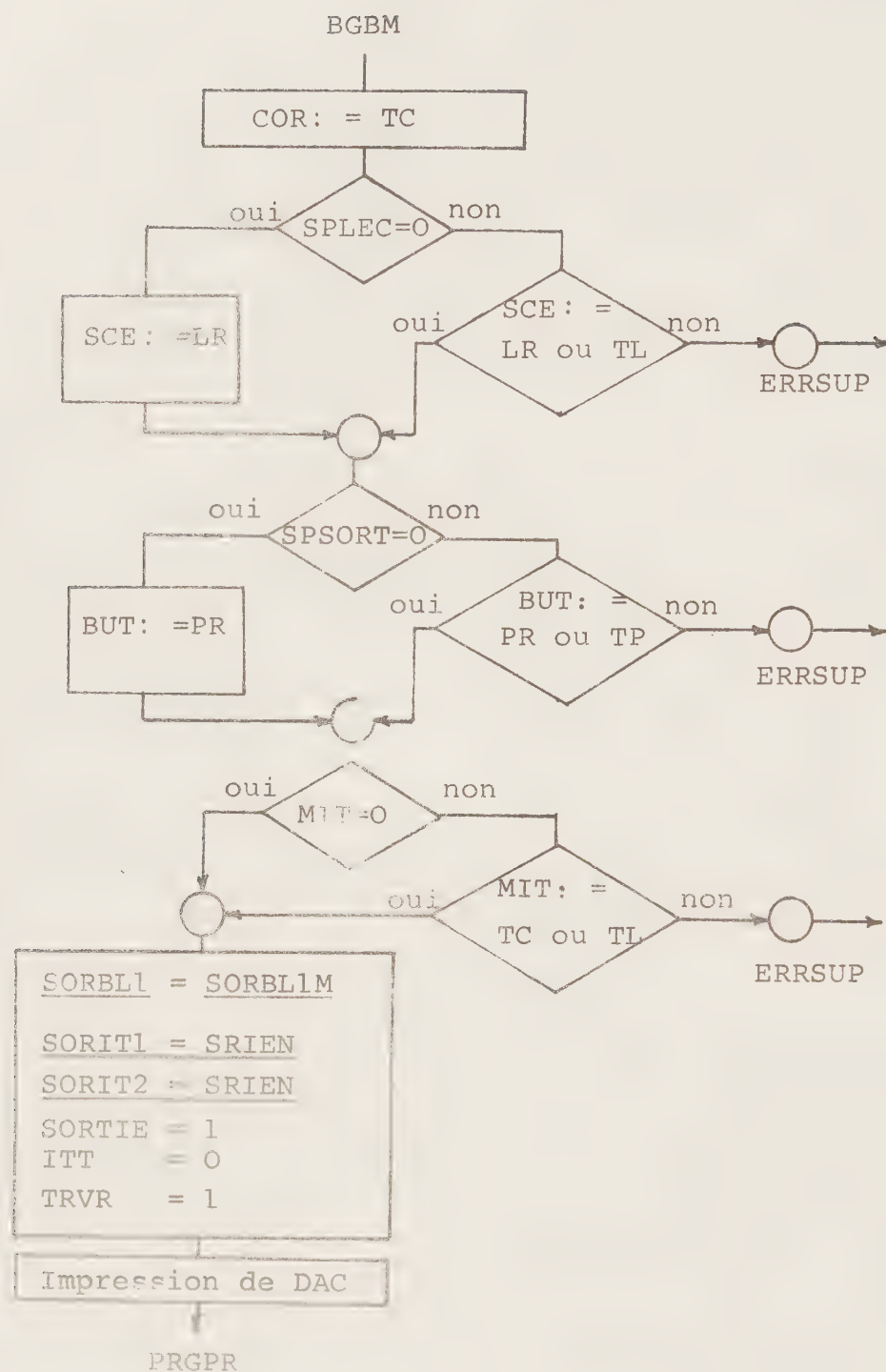
La détection du point virgule crée un branchement vers SUP1.

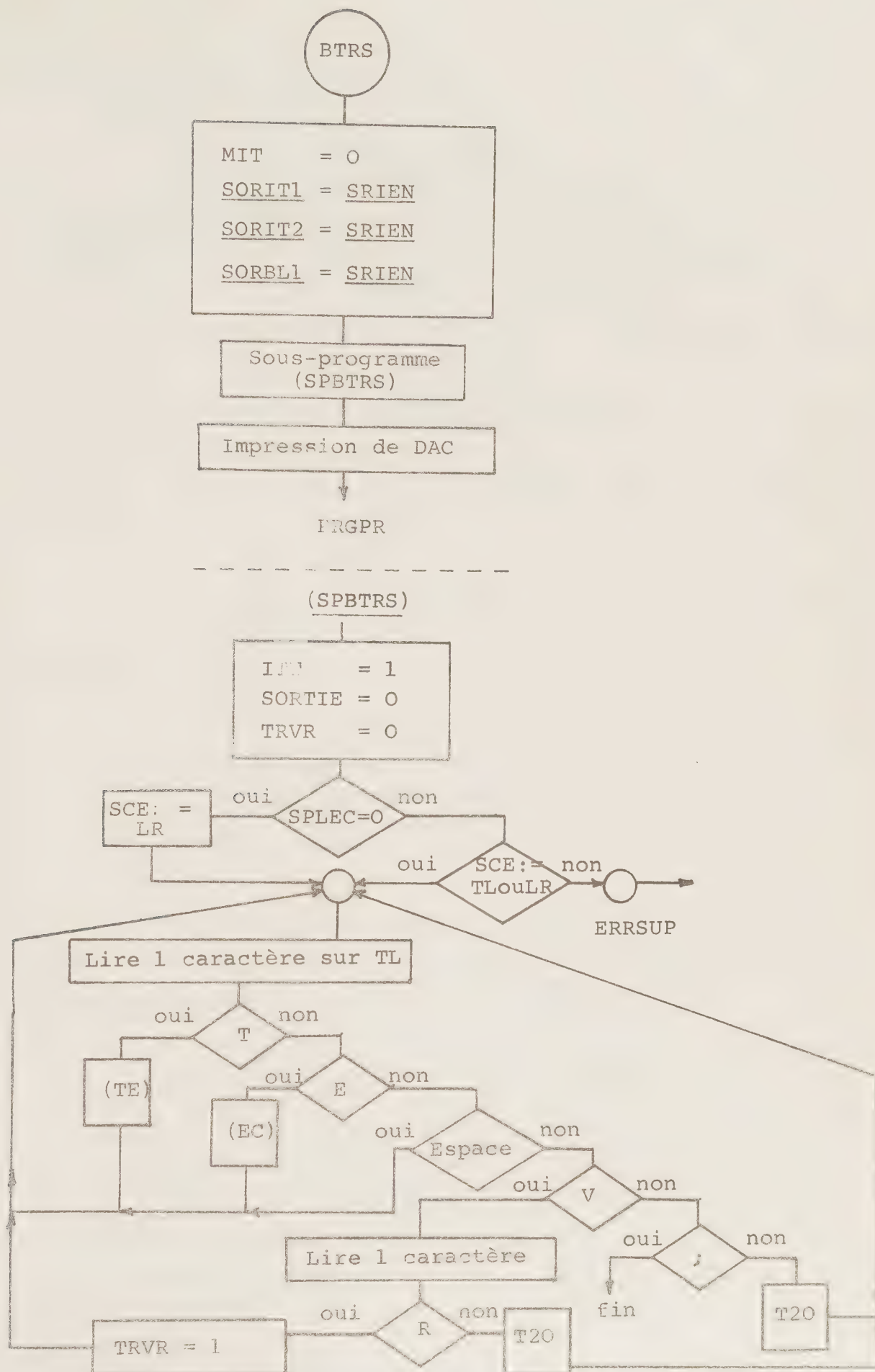
Les programmes BGBS, BGBM, BTRM, BTRS et BCOB contrôlent si les attributions de périphériques sont autorisées et attribuent aux fichiers non appelés leurs périphériques standards. Ils initialisent ensuite un certain nombre de variables et lancent le processeur appelé.

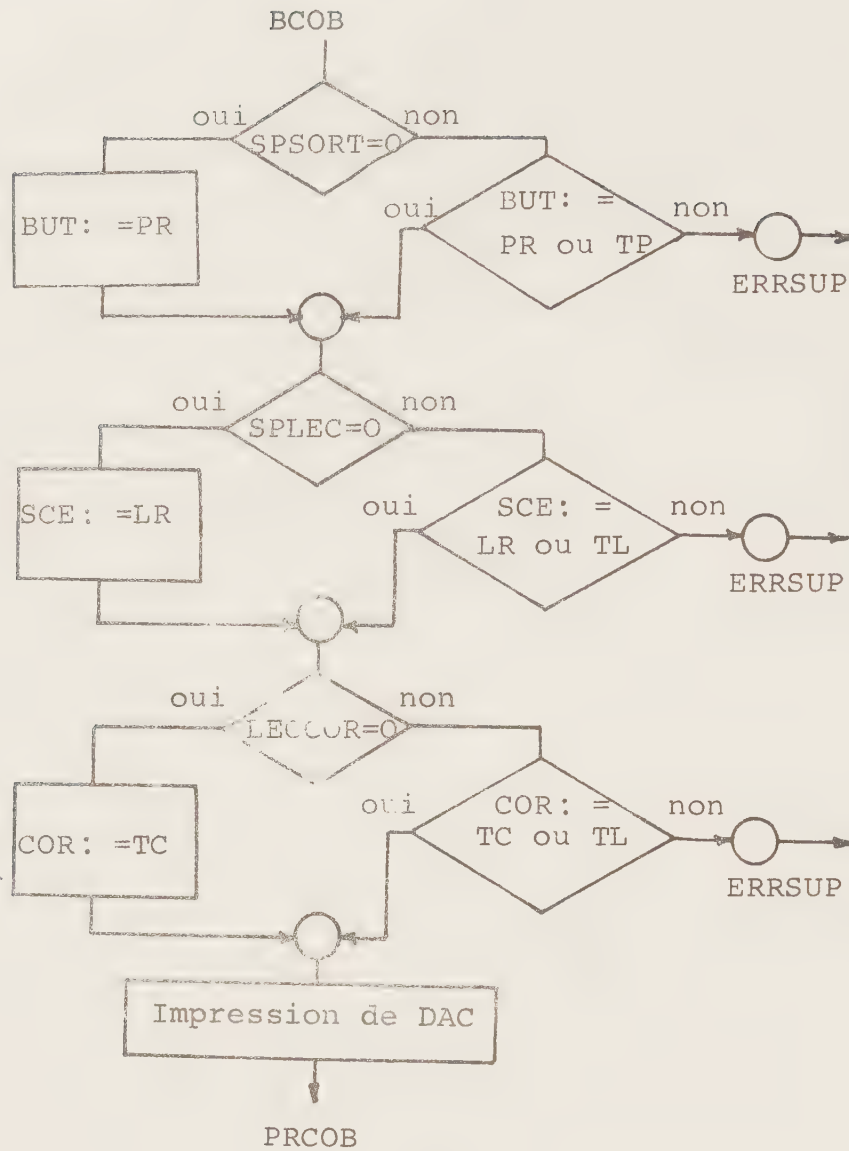
Le sous-programme SPBTRS utilisé par BTRS et BTRM permet, avant le départ des processeurs TRS et TRM, de programmer la vue et l'échelle choisies pour le tracé, et d'autre part, initialise la variable TRVR qui vaudra 1 si l'on désire obtenir le tracé des déplacements en vitesse rapide et 0 dans le cas contraire.











II - Programme TRM .

Ce processeur permet de tracer le profil de la pièce qui correspond à une bande perforée dans un format accepté par l'armoire de commande d'une machine outil. Il se compose d'un programme principal (PRTRM) et de sous-programmes correspondant chacun à un mot différent relatif à la programmation manuelle.

II-1 Le programme principal PRTRM :

Ce programme est lancé en frappant la commande GO au clavier du téléimprimeur.

Il lit la bande source jusqu'au caractère % qui indique la fin des commentaires et le début du programme proprement dit.

Il lit ensuite 1 caractère, décode celui-ci et se branche vers le sous-programme relatif à ce caractère. Il faut noter qu'en programmation manuelle les mots sont toujours composés d'une lettre suivie de plusieurs chiffres dont le nombre dépend de la lettre.

Après l'exécution du sous-programme propre au caractère décodé, le programme PRTRM se rebranche à l'adresse TRM1., sauf pour le caractère fin de bloc (line feed).

II-2 Sous-programmes relatifs aux caractères F, M, S, N :

Ces sous-programmes examinent seulement l'écriture des nombres qui suivent les caractères concernés. Les valeurs de ces grandeurs ne servent en aucun cas au tracé du profil de la pièce programmée.

II-3 Sous-programmes relatifs aux caractères X, Y, Z, R :

Leur rôle est de lire la valeur de chaque coordonnée et de la mettre en mémoire en absolu, après avoir tenu compte du mode de cotation sous lequel elles sont apparues.

Pour la coordonnée R, qui exprime un déplacement en Z en vitesse rapide, le nombre lu est rangé en Z et la vitesse d'avance est positionnée en "VR" (vitesse avance rapide non rémanente).

II-4 Sous-programme relatif au caractère G :

Les fonctions G peuvent exprimer, soit :

- des cycles d'usinage
- le mode de cotation
- le mode de déplacement

Ce sous-programme examine le nombre qui suit la lettre G et suivant la valeur de celui-ci initialise le cycle (CYG) , la vitesse d'avance (MODVA) ou le mode de cotation(COT).

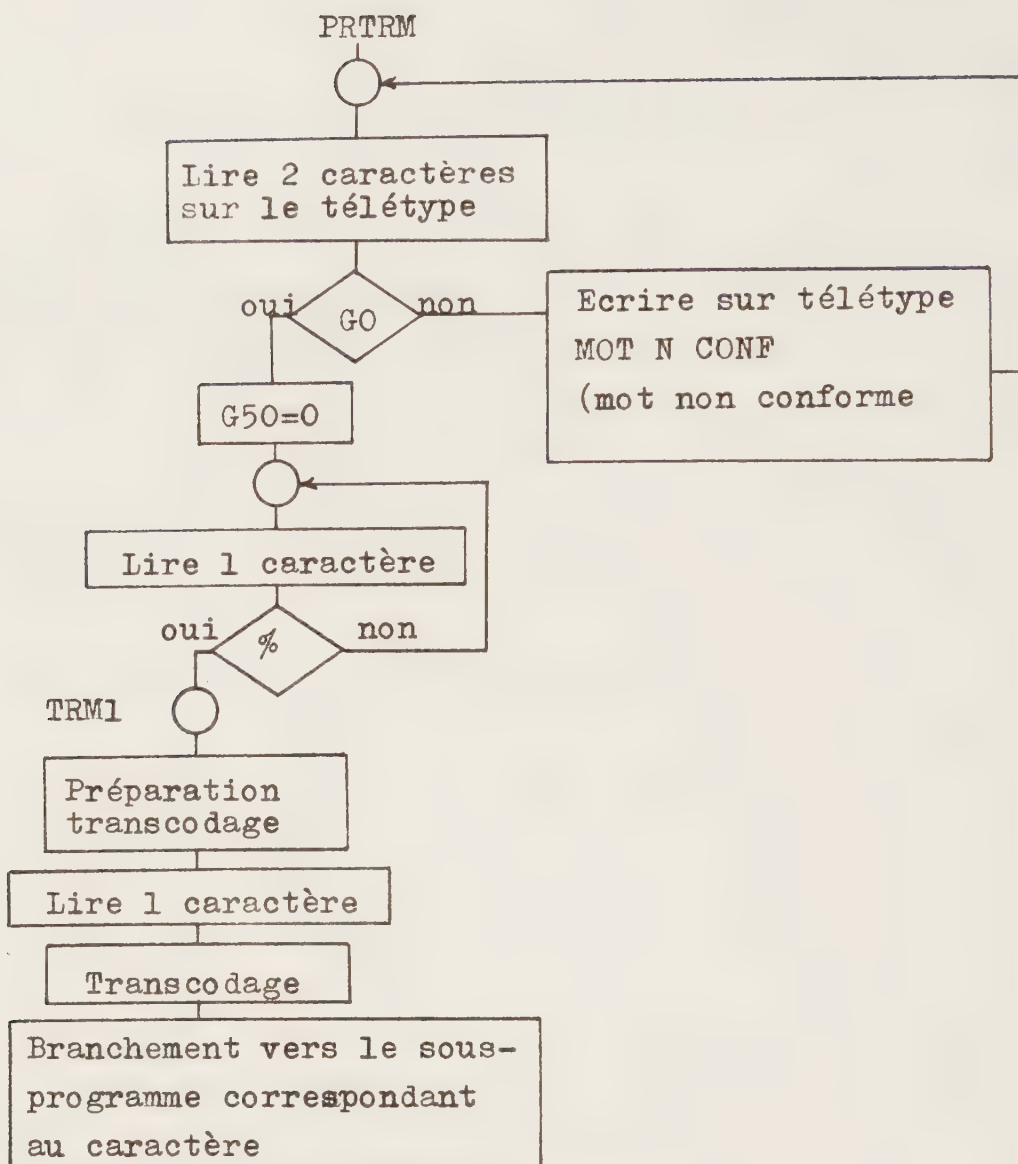
II-5 Sous-programme FBLOC :

Lorsque le caractère line feed apparaît, la machine à commande numérique exécute le déplacement indiqué dans le bloc. Dans le cas présent, c'est le sous-programme TRBLO qui commande ce déplacement sur la table traçante.

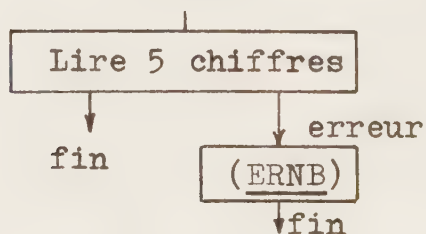
Si le mot G50 est apparu dans le bloc, il ne faut pas l'exécuter car ce mot exprime un changement d'axe. Les coordonnées qui apparaissent sont les nouvelles coordonnées du point précédemment programmé par rapport à la nouvelle origine.

ANNEXE 9.

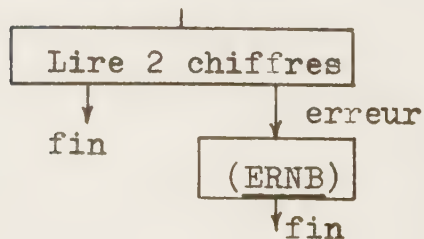
Il faudra donc, si G50 est apparu, donner aux variables XT et YT, qui expriment les coordonnées de la plume, les valeurs qui apparaissent dans le bloc, sans oublier de tenir compte de la vue qui a été programmée.



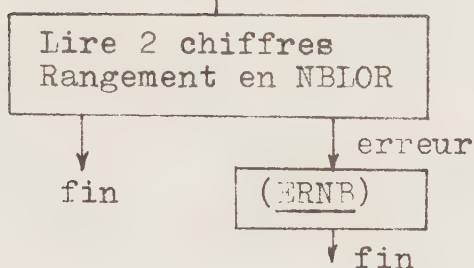
F (EFE)



M ou S (ESE)

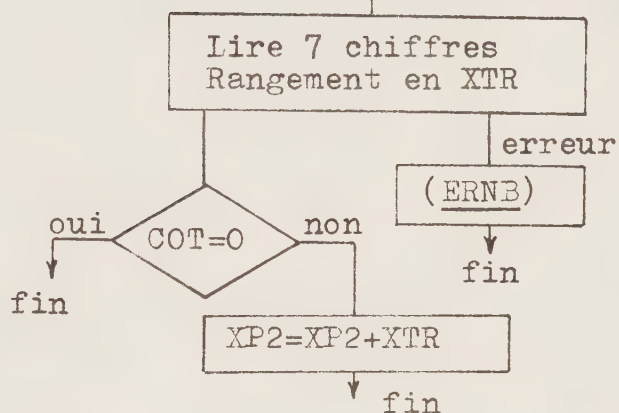


N ou : (DEUXP)

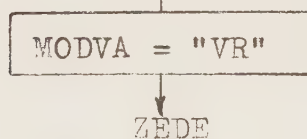
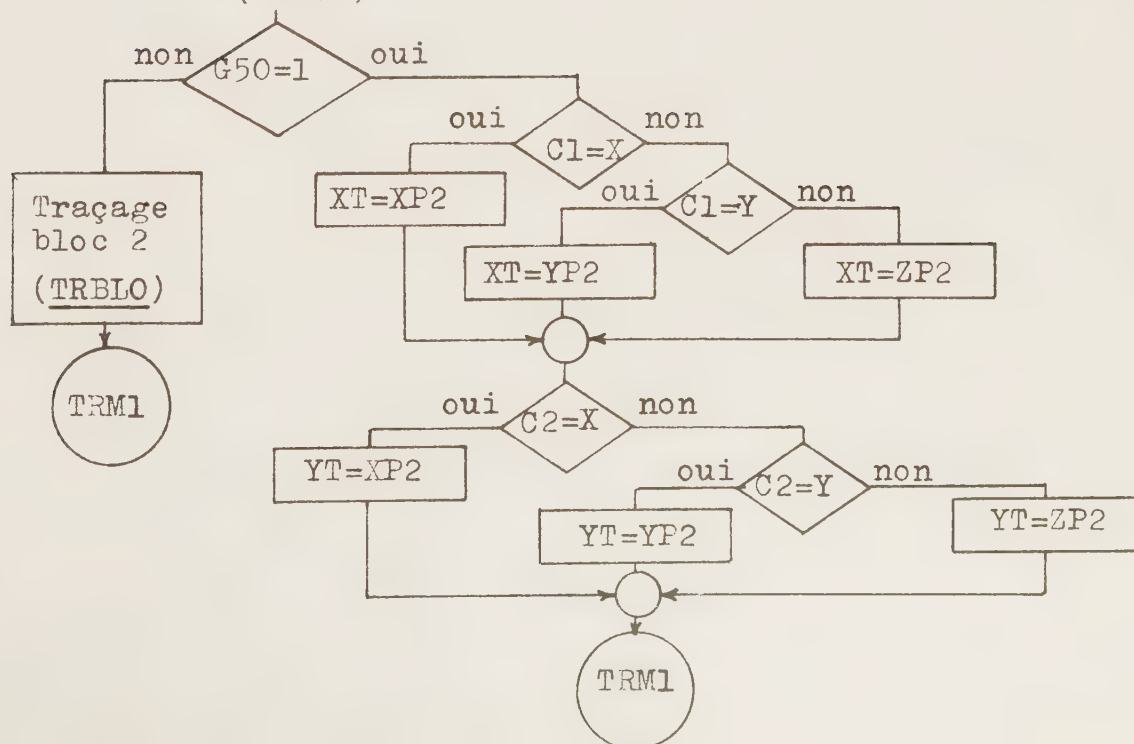


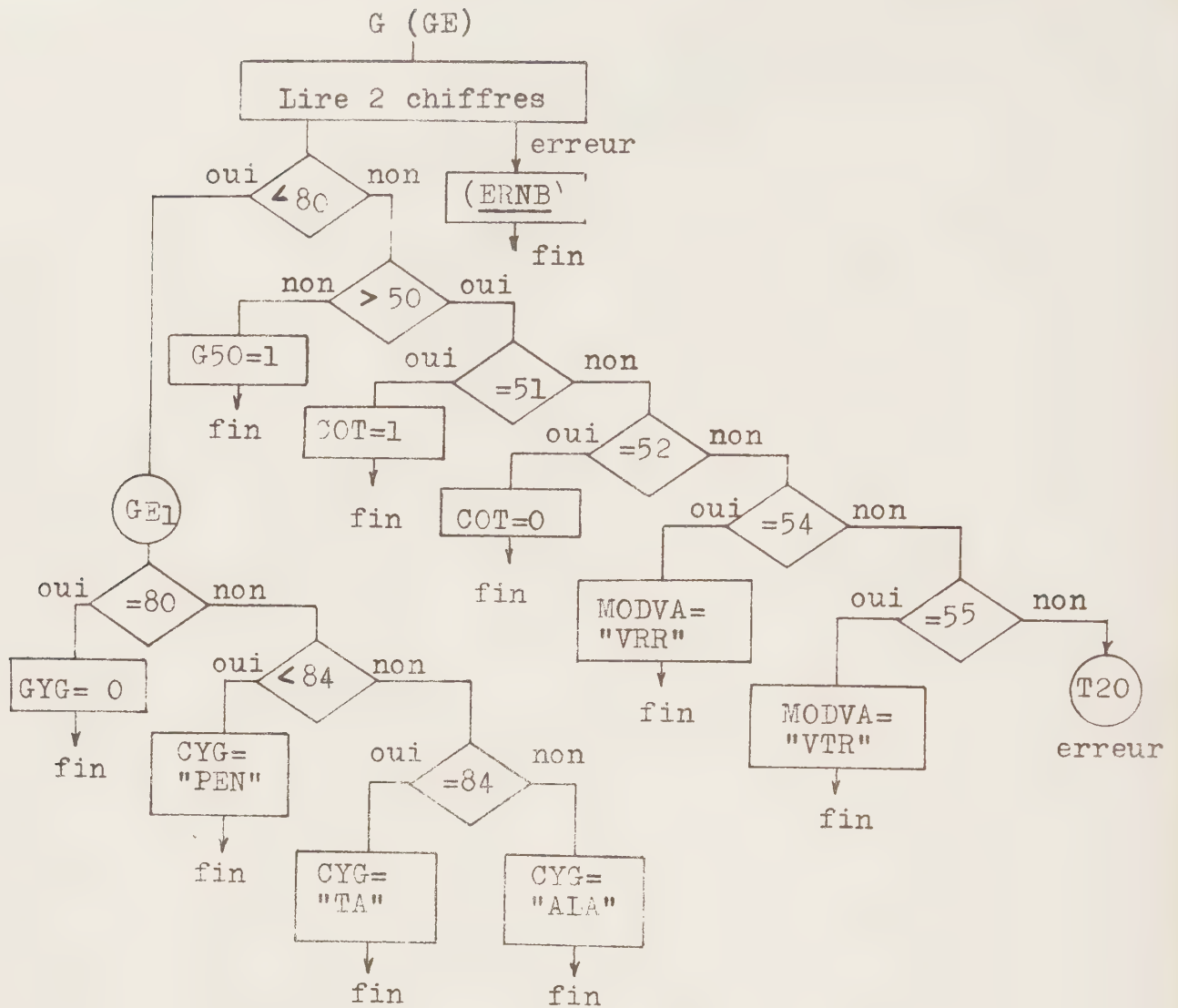
X (IXE)

De même pour
Y (IYE)
Z (ZEDE)



R (ERE)

Line feed
(FBLOC)



III - Programme COB .

Ce processeur a pour but de faciliter la correction des bandes perforées aussi bien en symbolique qu'en format ''machine''.

C'est un programme autonome qui n'utilise comme sous-programmes que ceux qui concernent les entrées-sorties.

Pour lancer ce processeur, il faut frapper la commande GO au clavier du téléimprimeur.

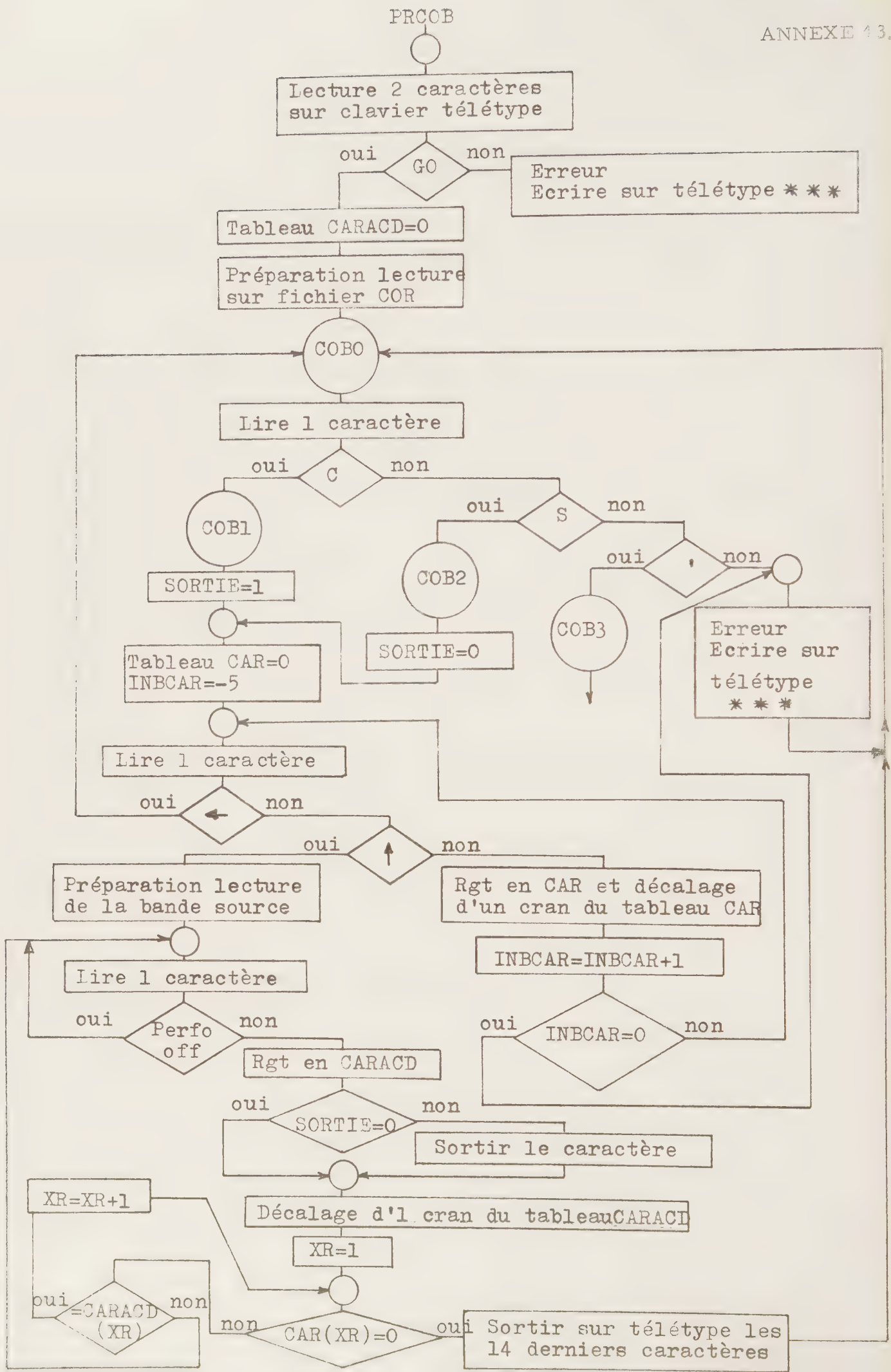
Les trois commandes acceptées par le programme COB sont composées d'un seul caractère. Lorsque celui-ci apparaît il est décodé et le programme bifurque vers les adresses COB1 , COB2 ou COB3 suivant la commande lue.

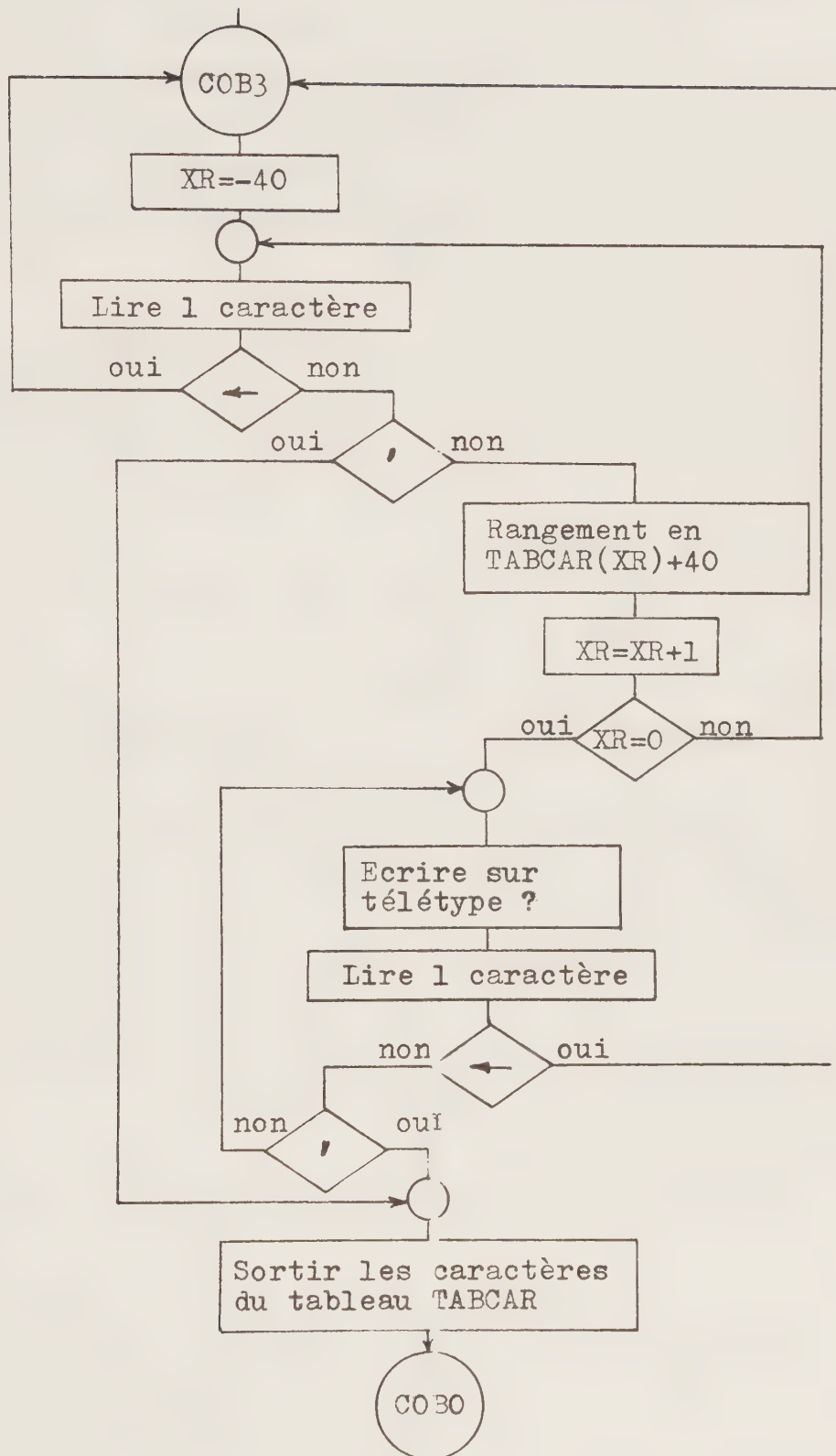
Les commandes C et S sont suivies d'un certain nombre de caractères qui vont permettre l'arrêt de la bande source.

Ces caractères sont rangés dans le tableau CAR et ensuite comparés aux caractères du tableau CARACD lus sur la bande source.

La variable sortie est mise à 0 par la commande S et à 1 par la commande C et conditionne la recopie ou non de la bande source.

Le caractère ' permet de délimiter une phrase à insérer sur la bande objet. Il est traité différemment s'il se trouve au début (COB3) ou à la fin de l'insertion. Celle-ci peut présenter au maximum 40 caractères qui sont rangés dans le tableau TABCAR.





IV - Les processeurs TRS , GBS , GB M .

Les trois processeurs TRS, GBS, GBM sont un seul et unique programme dont les sous-programmes d'entrée et de sortie différent et sont initialisés respectivement par les programmes BTRS, BGBS, BGBM.

Dans les organigrammes, les variables soulignées précisent quels sont les sous-programmes de sortie utilisés. Ainsi pour BGBM les égalités

SORIT1 = SRIEN
SORIT2 = SRIEN
SORBL1 = SORBL1M

signifient que les sous-programmes SORIT1 et SORIT2 qui assurent normalement la sortie des informations technologiques symboliques, sont remplacés par le sous-programme SRIEN qui n'effectue rien.

D'autre part, le sous-programme de sortie des informations géométriques SORBL1 devient SORBL1M (sous-programme de sortie des informations machine).

IV-1 Le programme PRGPR :

Ce programme, après une initialisation du mode de cotation et de vitesse avance, teste la valeur de la variable MIT. Si celle-ci est nulle, c'est qu'aucune modification d'informations technologiques n'est présente sur le fichier MIT, sinon il faudra lire sur ce dernier le numéro de la première section qui y apparaît et le ranger en NSMIT.

Après une autre série d'initialisations, on passe à la lecture du fichier source et le premier caractère lu est transcodé. Le résultat de ce transcodage étant l'adresse du sous-programme de traitement du caractère en question.

Nous n'explicitons pas ici tous les sous-programmes de traitement des divers caractères lus, les organigrammes correspondants suffisent. Cependant, les sous-programmes présentant des particularités importantes sont décrits succinctement ci-dessous.

IV-2 Sous-programme TE :

Ce sous-programme est lancé par la lecture de la lettre T sur le fichier source.

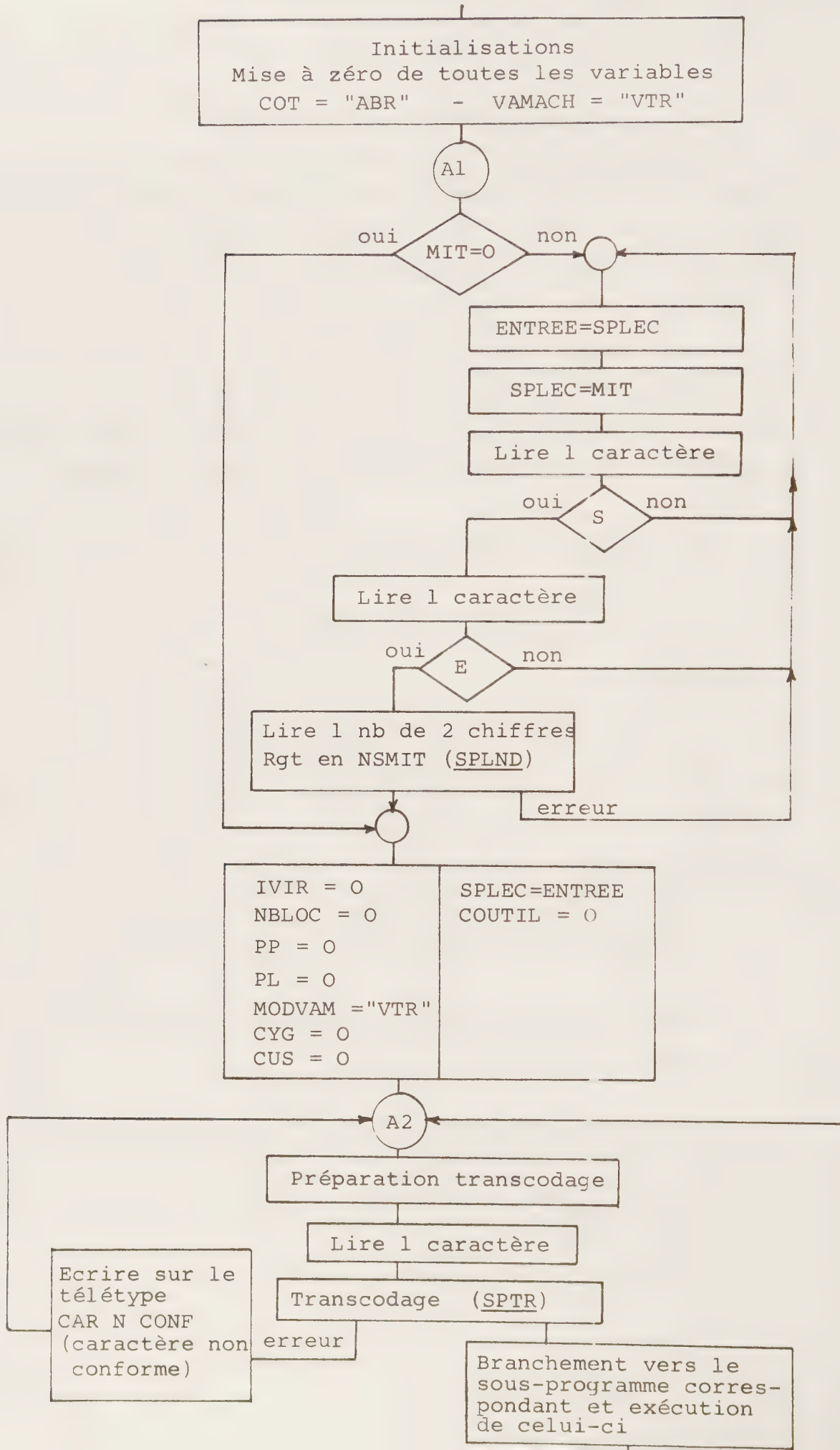
Les mots commençant par T peuvent être TS, TA ou TT suivi de 2 caractères distincts pris parmi X , Y , Z. Notons qu'il y a un contrôle en T16 de la définition préalable de l'échelle avec laquelle on désire effectuer le tracé.

IV-3 Sous-programme VT : _

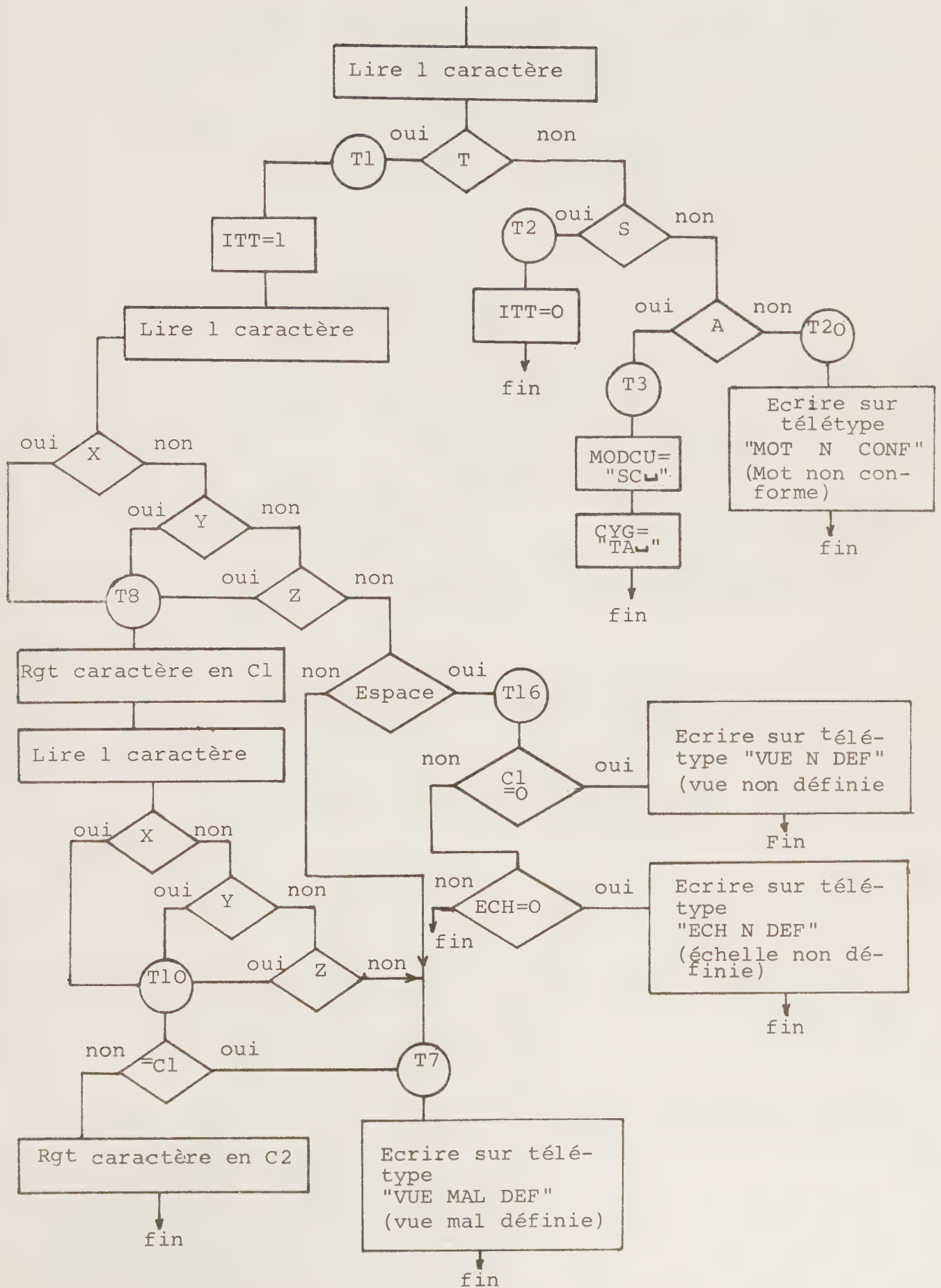
Traitement du caractère V par lequel débutent les mots VA , VRP ou VRN .

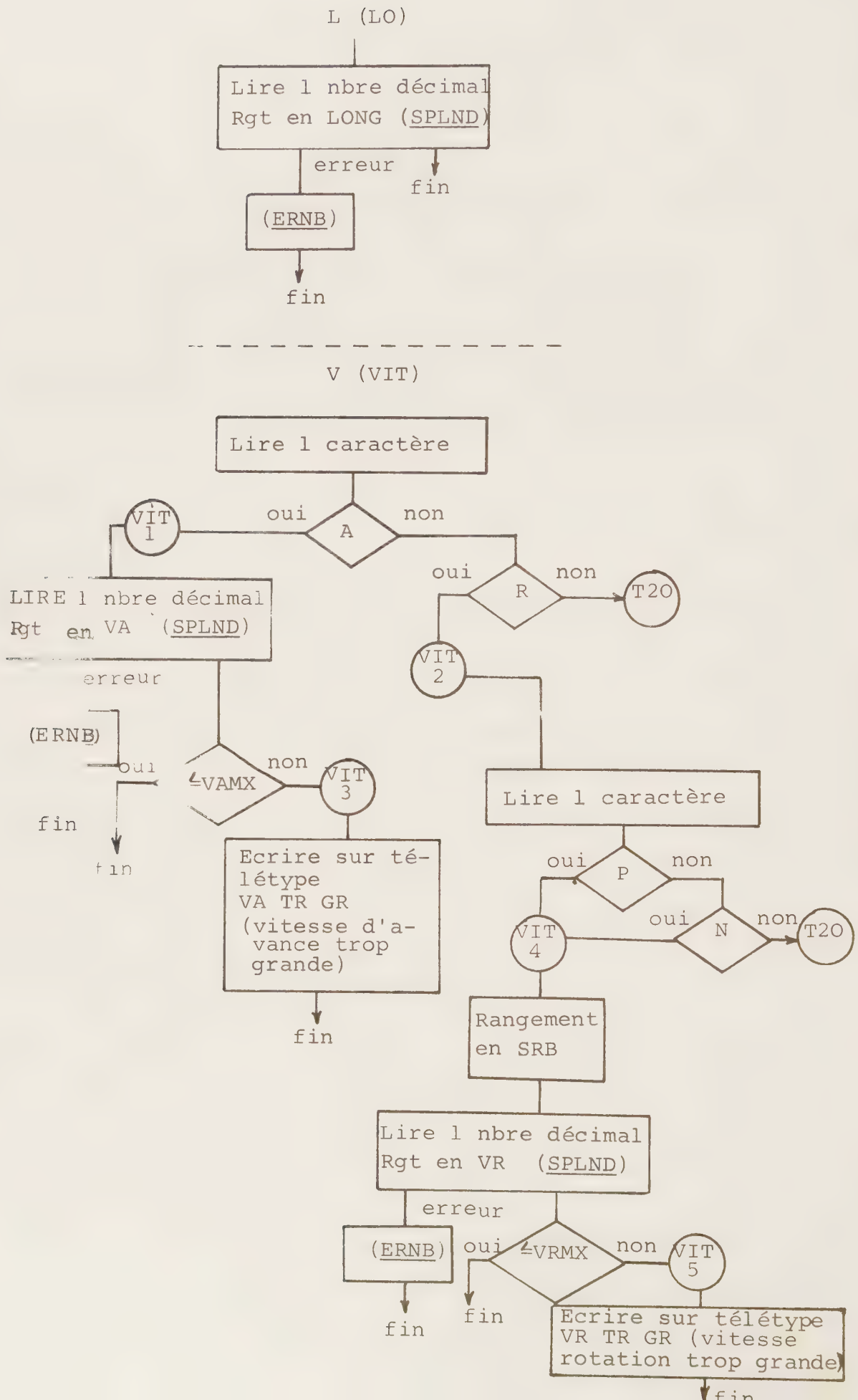
Au cours de ce sous-programme une comparaison avec les vitesses maximales admissibles par la machine est effectuée et un message d'erreur est imprimé en cas de dépassement.

(Programme principal :
processeurs GBS, GBM, TRS)



T (TE)





IV-4 Sous-programme SE :

Au cours de ce sous-programme correspondant à la lettre S , si le fichier MIT a été préalablement initialisé ($MIT \neq 0$), on effectue une comparaison du numéro de la section en cours à la valeur de NSMIT. S'il y a concordance, les informations technologiques seront lues sur le fichier MIT et non sur le fichier source.

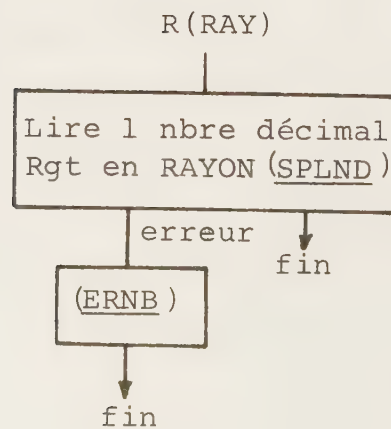
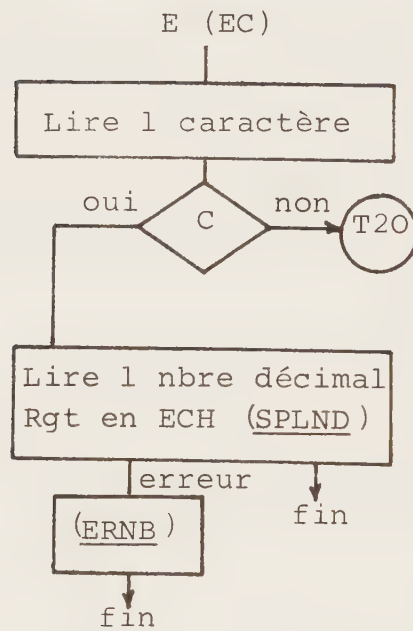
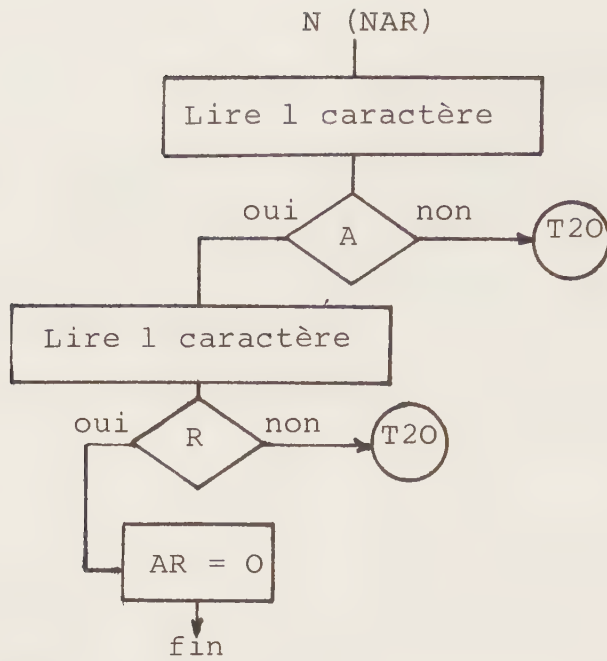
IV-5 Sous-programme VIR :

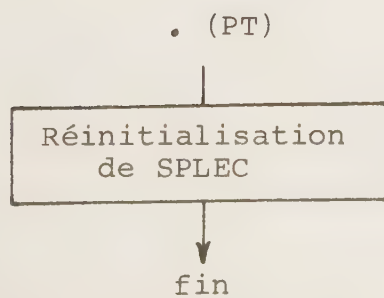
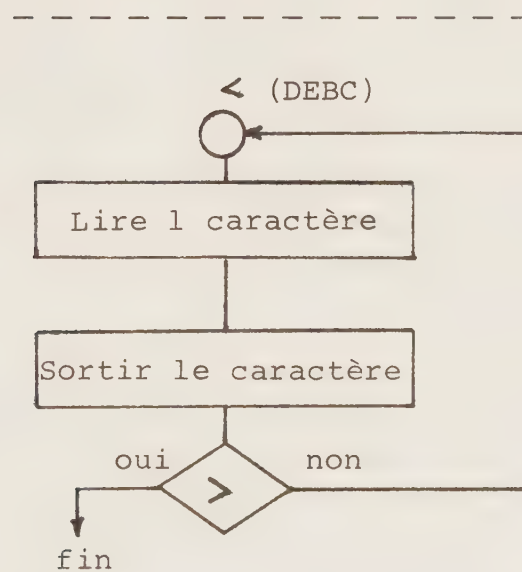
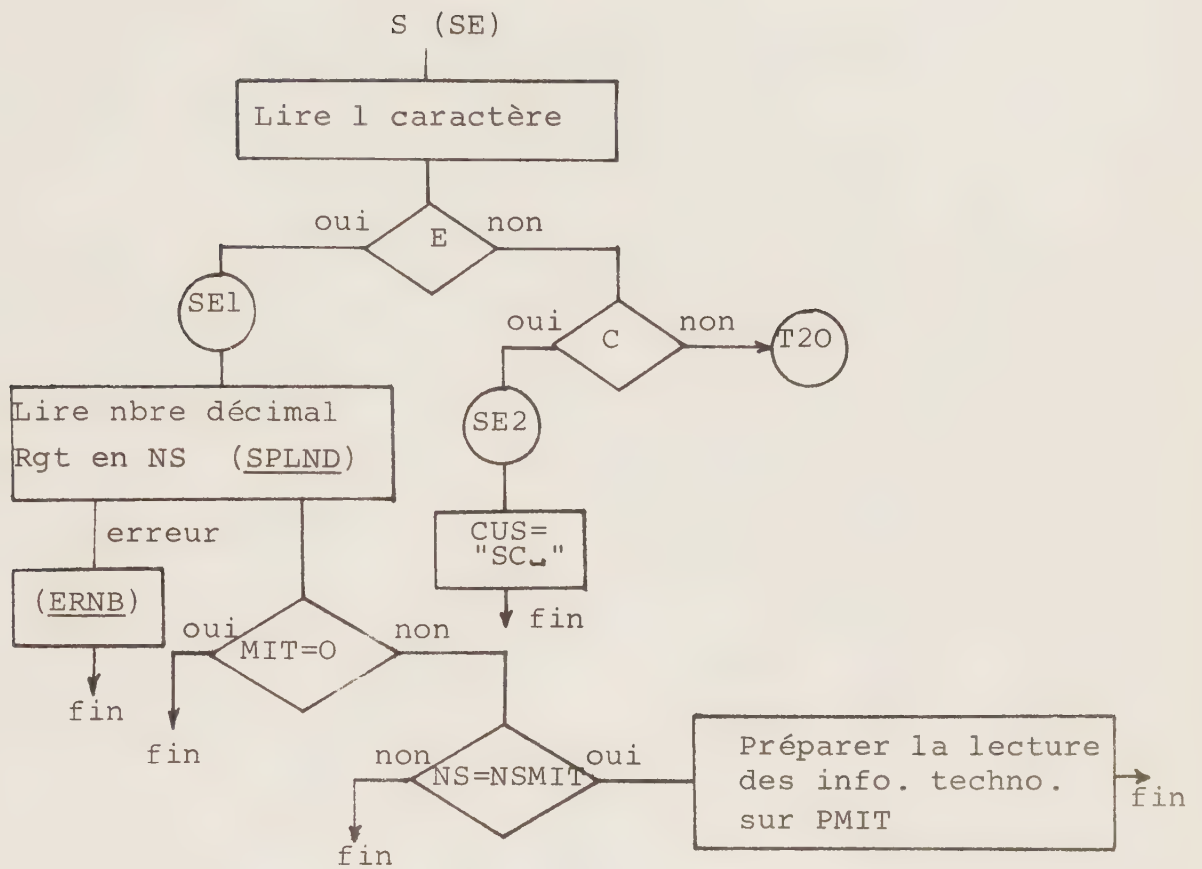
Une virgule dans la phrase d'informations technologiques exprime la fin d'un groupe de données. Si le processeur appelé est GBS (SORTIE = 0) , il faut contrôler les informations technologiques (CONTIT) et les restituer sur la bande objet. S'il s'agit de la première virgule apparue dans la phrase le sous-programme de sortie (SORIT1) sera différent du sous-programme SORIT2 qui entraîne la perforation sur la bande objet les données technologiques apparaissent après une virgule.

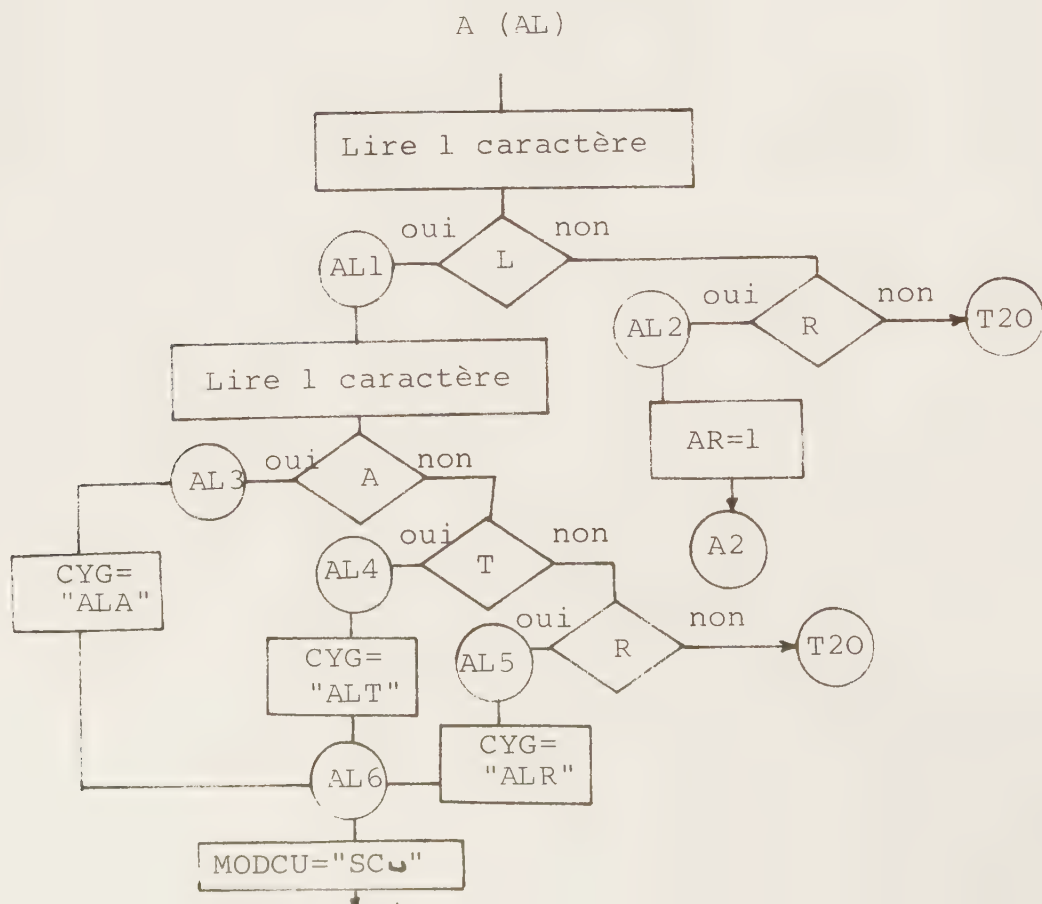
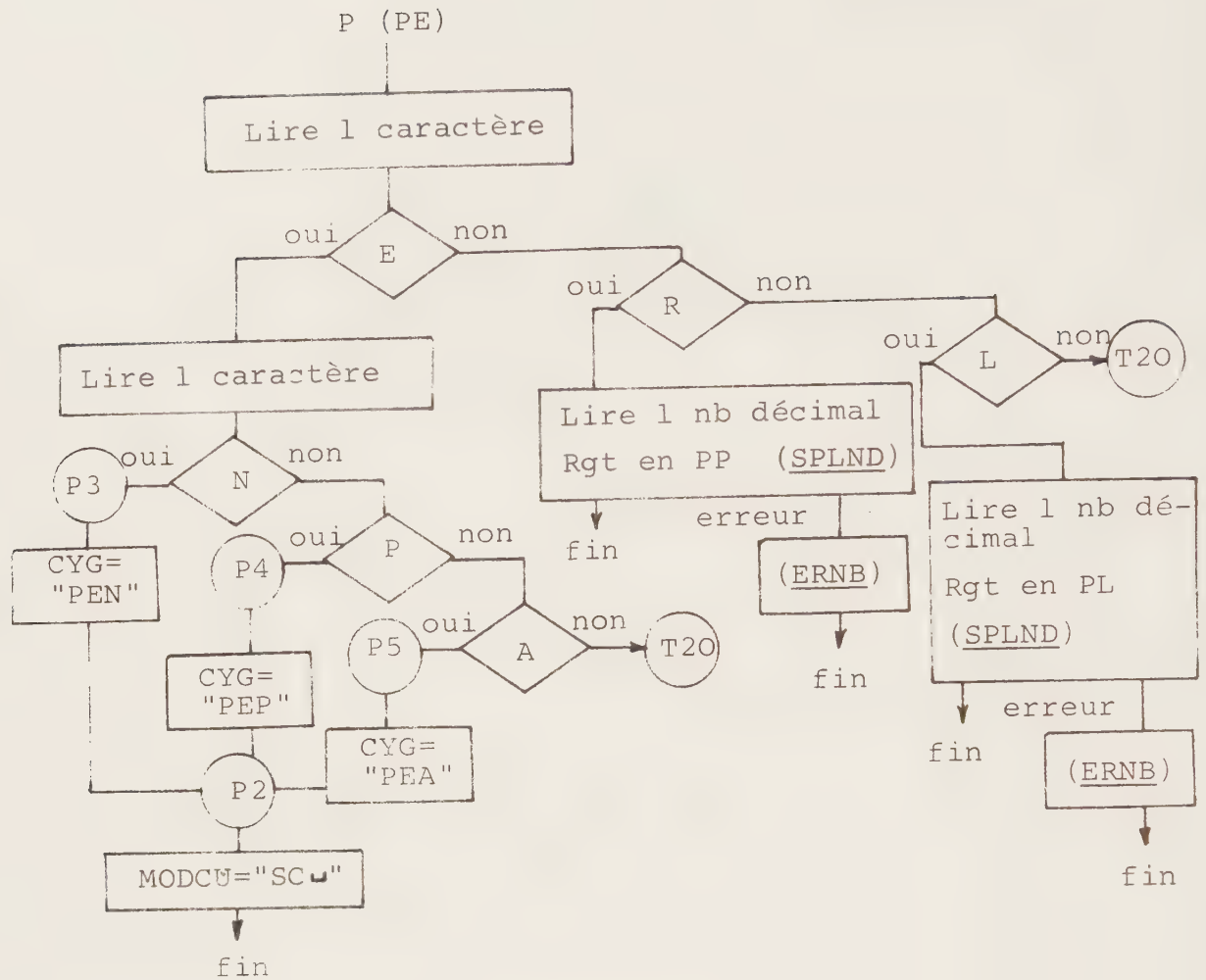
Si le processeur appelé est GBM, les données précédant la virgule sont prises en compte et la bande objet est lue jusqu'au prochain point virgule délimitant la phrase de données technologiques.

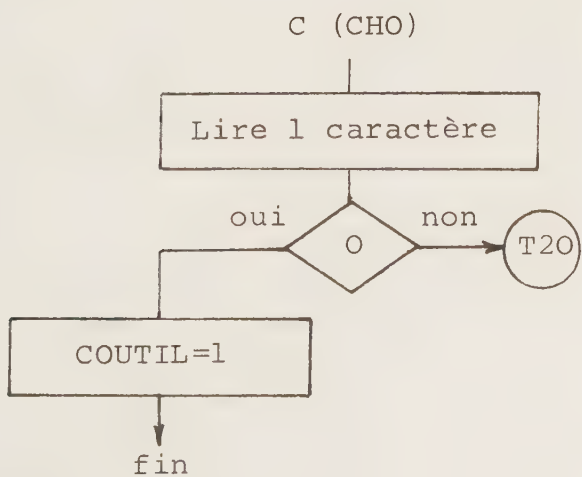
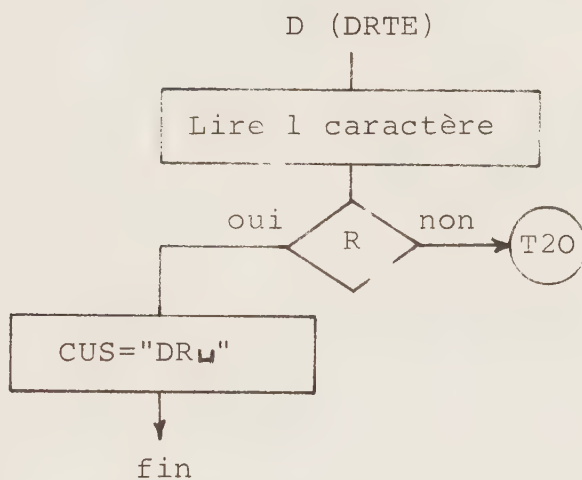
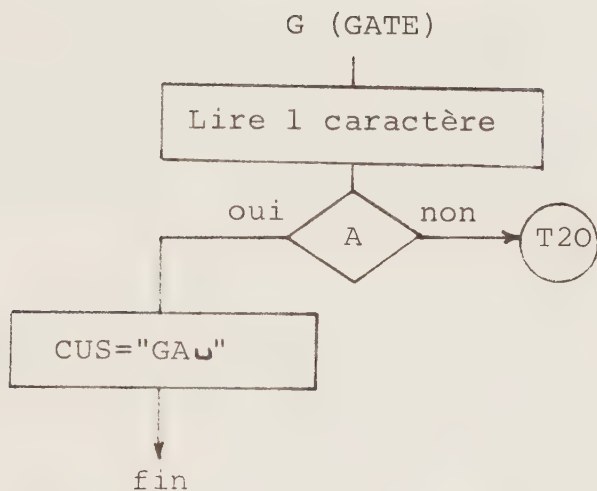
IV-6 Sous-programme PV :

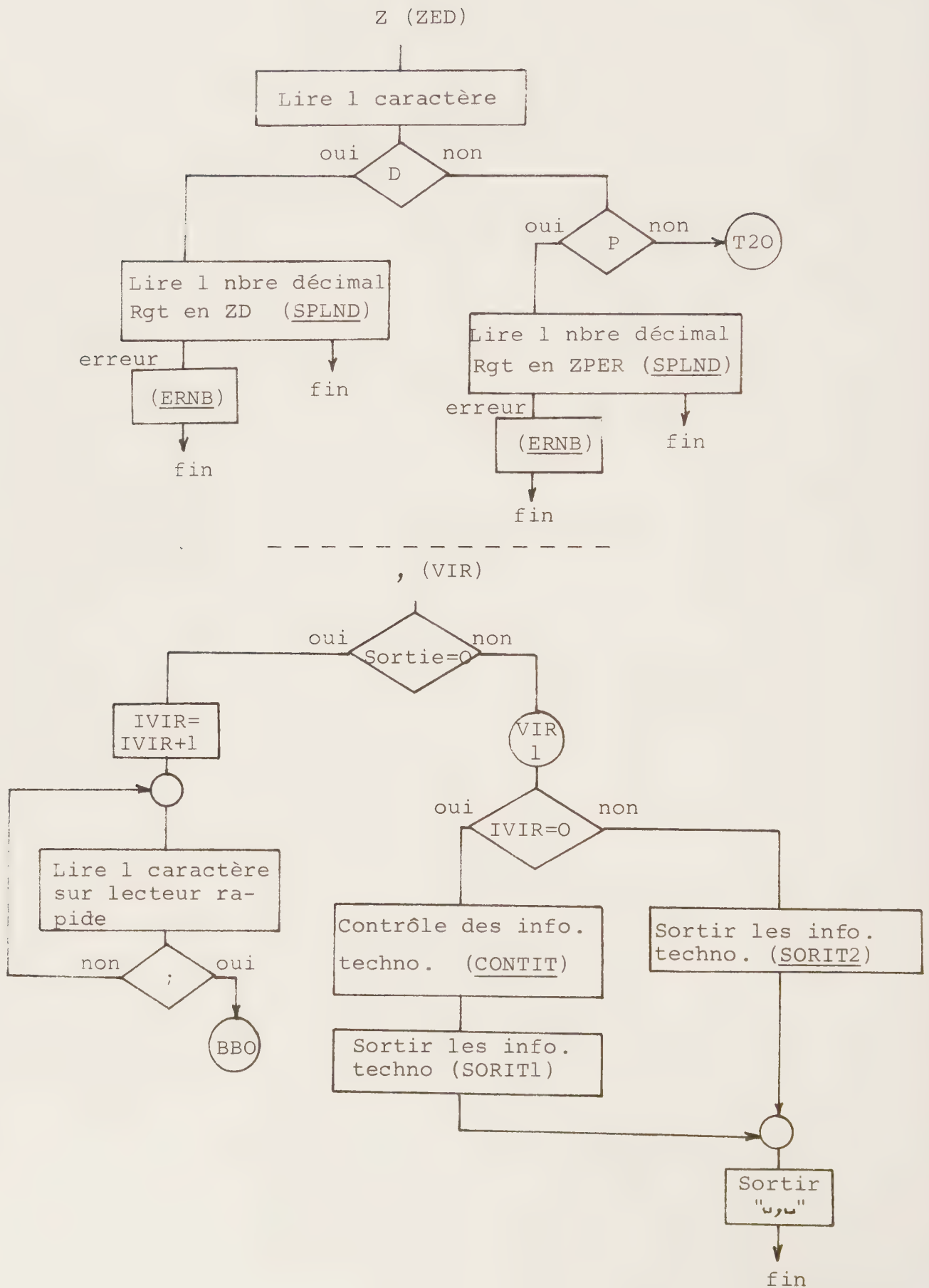
Le point virgule indique la fin des données technologiques. Si la variable SORTIE vaut 0 , il s'agit du processeur GBS, on restitue alors la phrase des grandeurs technologiques après contrôle. Sinon la variable IVIR est initialiser à la valeur 1000 pour indiquer que tous les groupes de données ont été lus et qu'il faut passer à la section suivante (voir sous-programme FS).











S'il y avait des modifications d'informations technologiques (MIT 0) , le fichier source est remis en fichier d'entrée et la bande source lue jusqu'au prochain point virgule. Les grandeurs technologiques de la bande source sont donc ignorées.

Il faut alors lire un bloc d'informations géométriques, ce qu'effectue le sous-programme ENTINF. Le bloc 3 est ensuite transféré dans le bloc 2 et un branchement est effectué vers le programme de traitement de la fonction lue.

IV-7 Sous-programme ENTINF :

Ce programme d'entrée d'un bloc d'informations géométriques commence par la remise à zéro partielle du bloc 3 où vont être introduites les grandeurs lues.

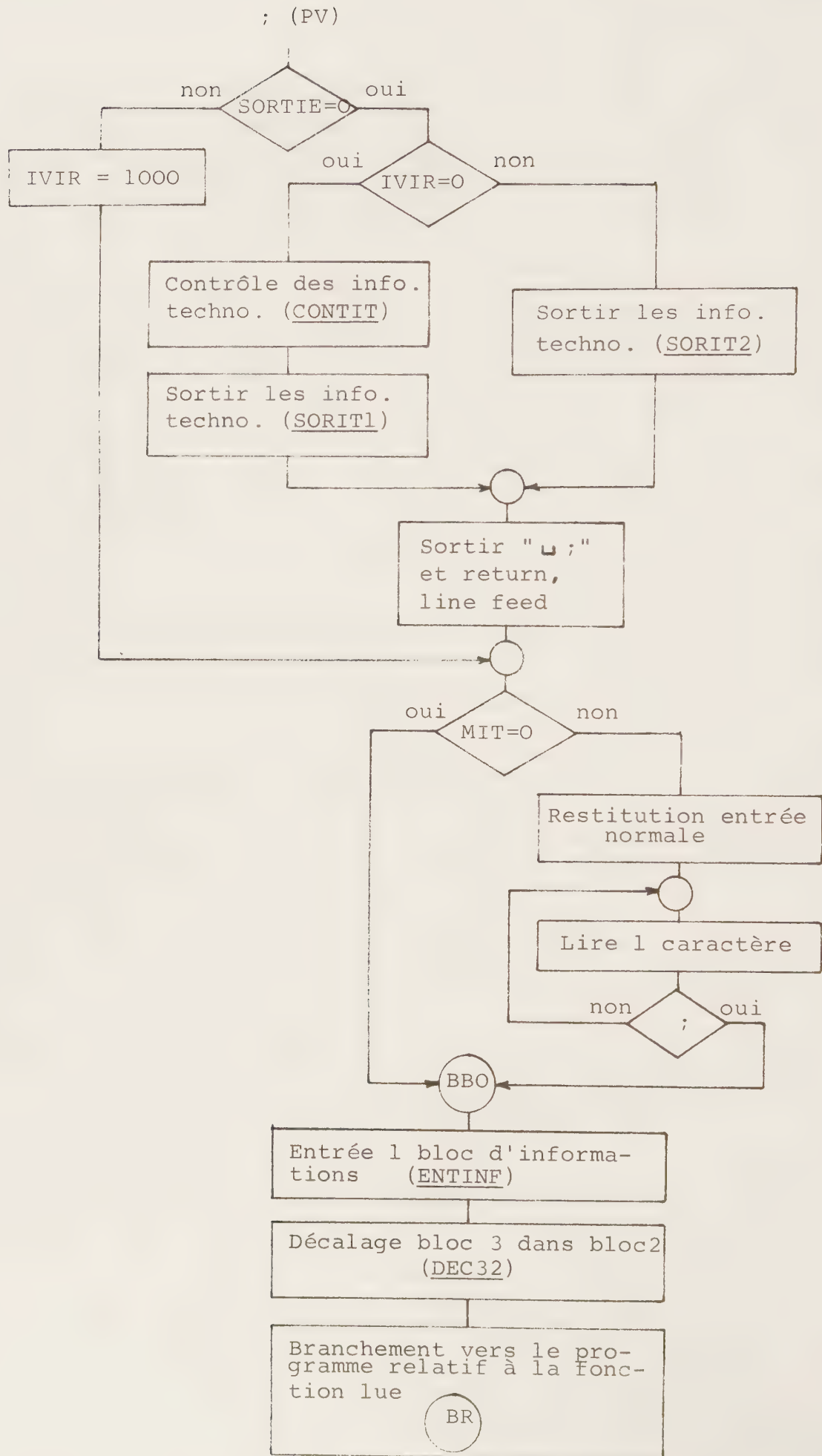
Après la préparation du sous-programme de transcodage, un caractère est lu et décodé, s'il s'agit d'un caractère conforme, un branchement est effectué vers le sous-programme correspondant.

Nous n'examinerons ici que les sous-programmes présentant des particularités intéressantes.

IV-8 Sous-programme FS :

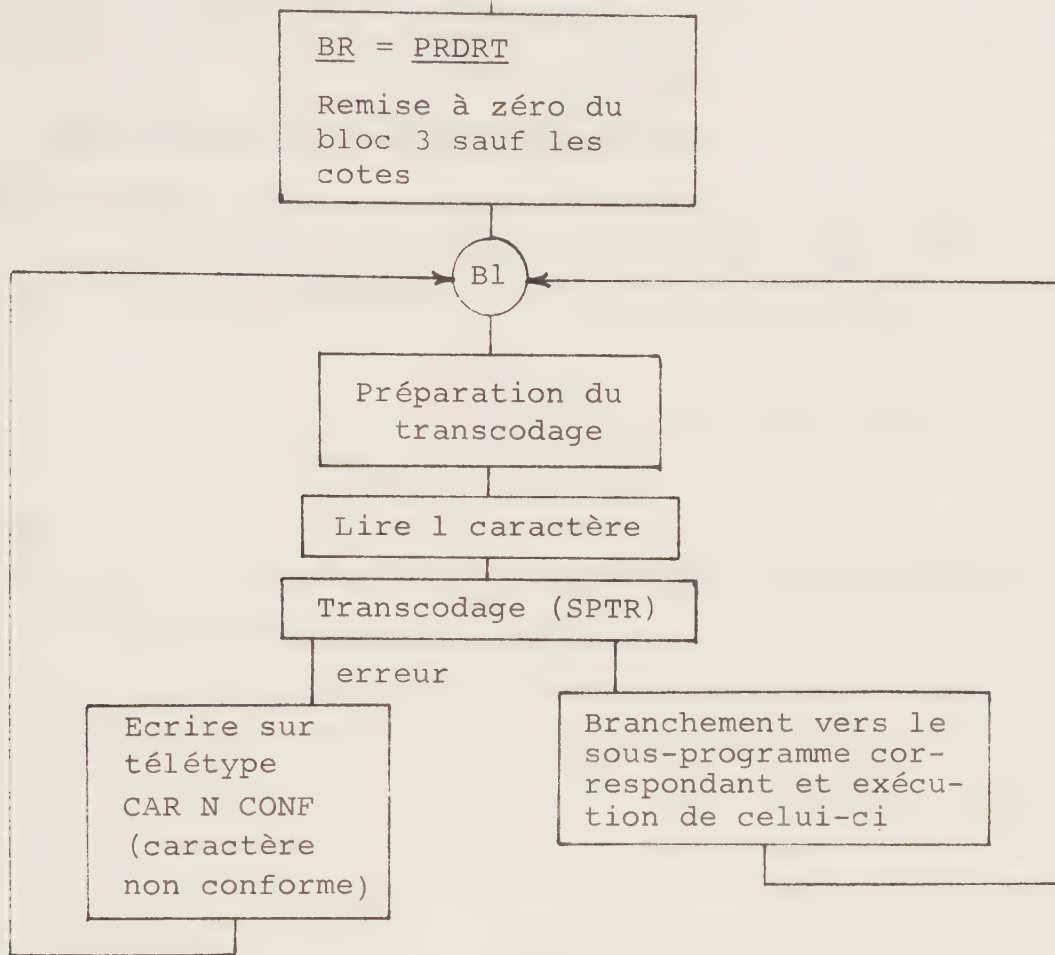
Ce sous-programme traite le caractère F donc les mots FS et FIN.

Pour FS (fin de section), s'il s'agit du processeur GBM (SORTIE 0) et si IVIR est différent de 1000 (phrase d'informations technologiques non totalement lue) , il y a retour en arrière de la bande source jusqu'au début de la section. D'autre part, suivant la

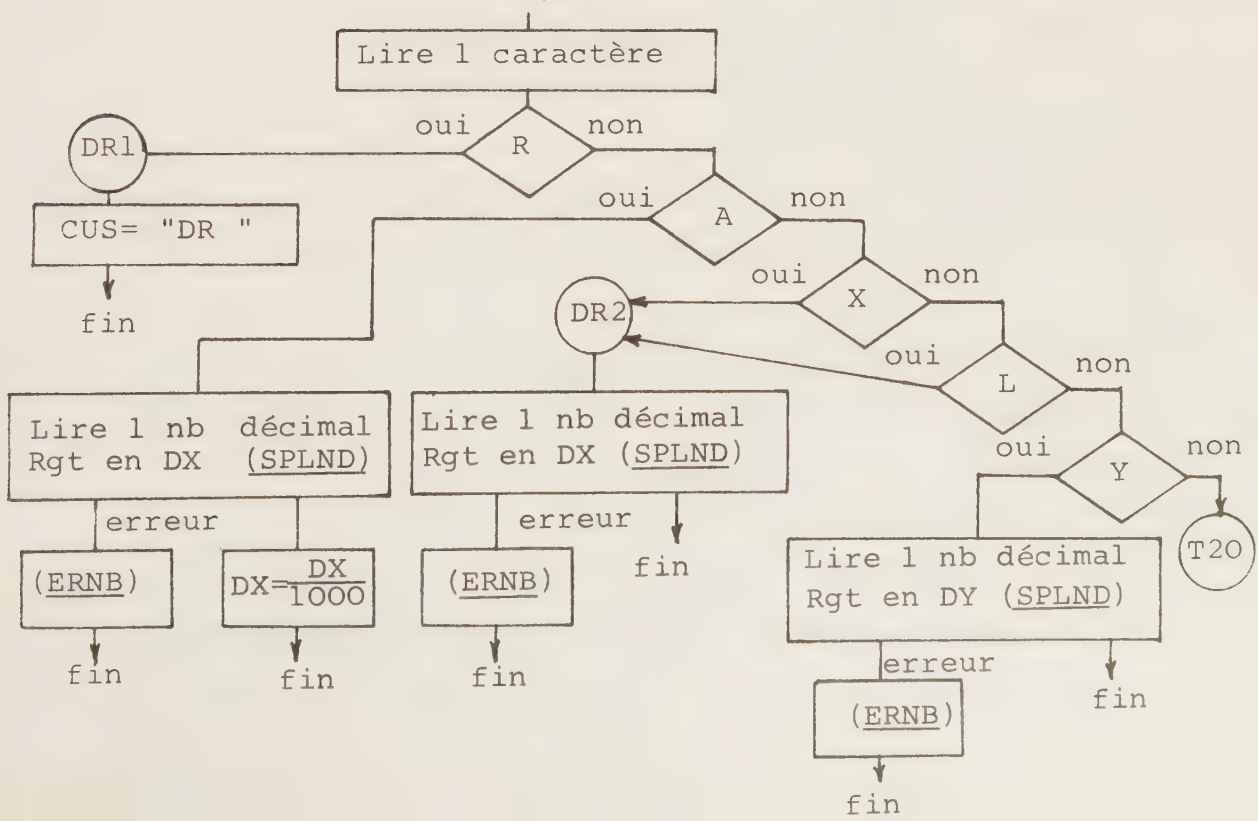


ENTINF

(Entrée d'un bloc d'informations)



D (DR)



valeur de MIT, il y aura, soit une lecture de virgules (n étant contenu dans IVIR) sur la bande source, soit la lecture du fichier MIT.

A l'apparition du mot FIN (fin du programme) on doit, soit restituer celui-ci (si SORTIE = 0), soit terminer la bande objet par les fonctions MO2 et M30 de fin de programme.

IV-9 Sous-programme DO :

Le caractère § indique le début d'une définition de fonction. Le sous-programme DO lit donc 3 caractères, les condense et recherche un correspondant dans le tableau des fonctions permises. A chacune de celles-ci correspond une adresse qui sera rangée en BR. La lettre qui suit la fonction désigne le sens de parcours sur la figure définie.

IV-10 Sous-programme TGT :

Il traite la lettre T par laquelle débute les mots TP, TPS ou TS, ceux-ci précisant si la figure décrite est tangente à la figure précédente, suivante ou aux deux.

Ces mots peuvent être suivis de modifications destinées à lever des indéterminations en cas de solutions multiples.

Ces modificateurs initialisent la variable TANG à des valeurs différentes suivant qu'il s'agit du premier modificateur ITGT = 0 ou du second s'il y en a un (ITGT ≠ 0).

Notons que ce sous-programme est utilisé également par le sous-programme CEE, à partir de TGTO, ce qui explique la présence d'une indexation de la variable TANG (XR = index).

IV-11 Sous-programme FB :

Ce programme est lancé par l'apparition d'un point virgule indiquant la fin d'un bloc d'informations.

Après une correction des coordonnées lues, en fonction du mode de cotation, le programme ENTINF prend fin.

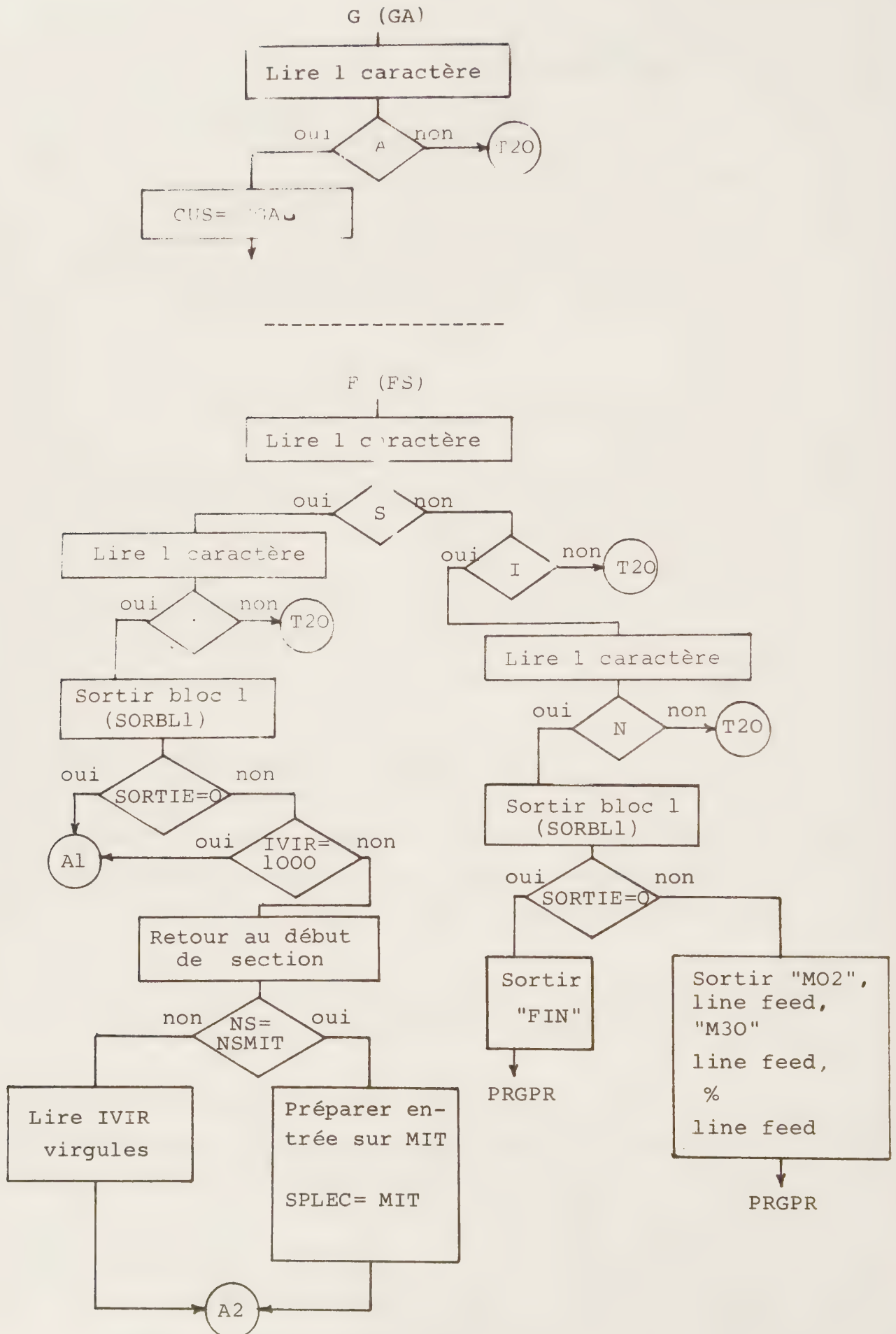
IV-12 Programmes BRP1 et BRP2 :

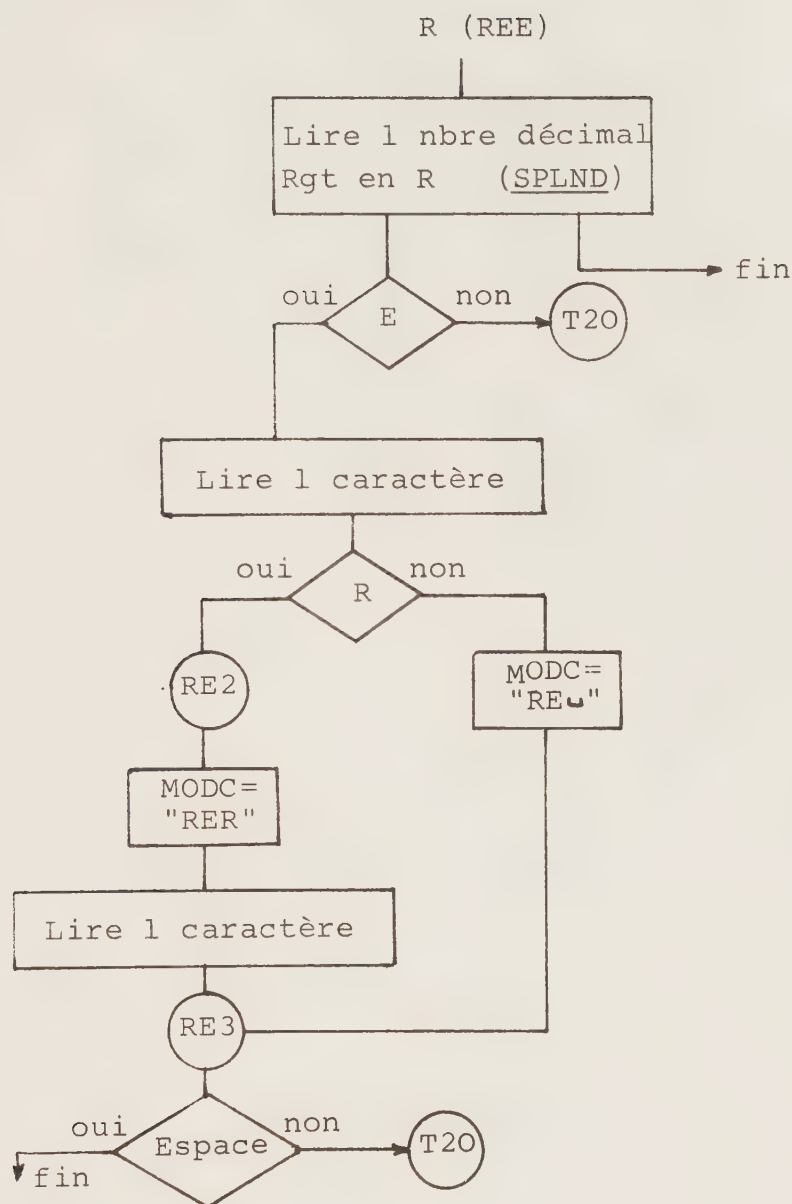
Ce sont en fait deux adresses de branchements distinctes du programme principal PRGPR.

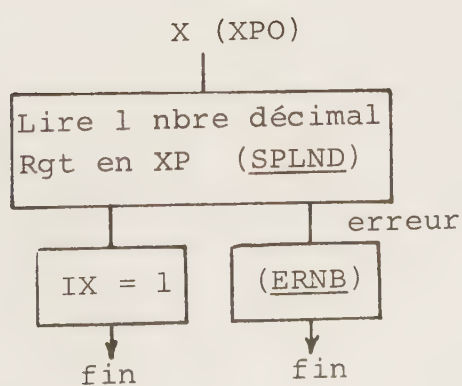
Si la fonction lue est définie par un seul bloc, le programme bifurque en BRP1 sinon en BRP2. En fait, ces deux parties de PRGPR assurent le tracé du ou des blocs d'informations géométriques lus. Après celui-ci il y a lecture d'un caractère, s'il s'agit d'un % (validation des informations), les blocs définissant la fonction sont perforés sur la bande objet et tous décalés d'un cran. Si le caractère lu est différent du % la plume de la table traçante est remise à sa position précédente et le bloc PR1X (Préparation du bloc 1) est rechargé par le bloc 1.

IV-13 Sous-programmes TRBLO et BPLCO :

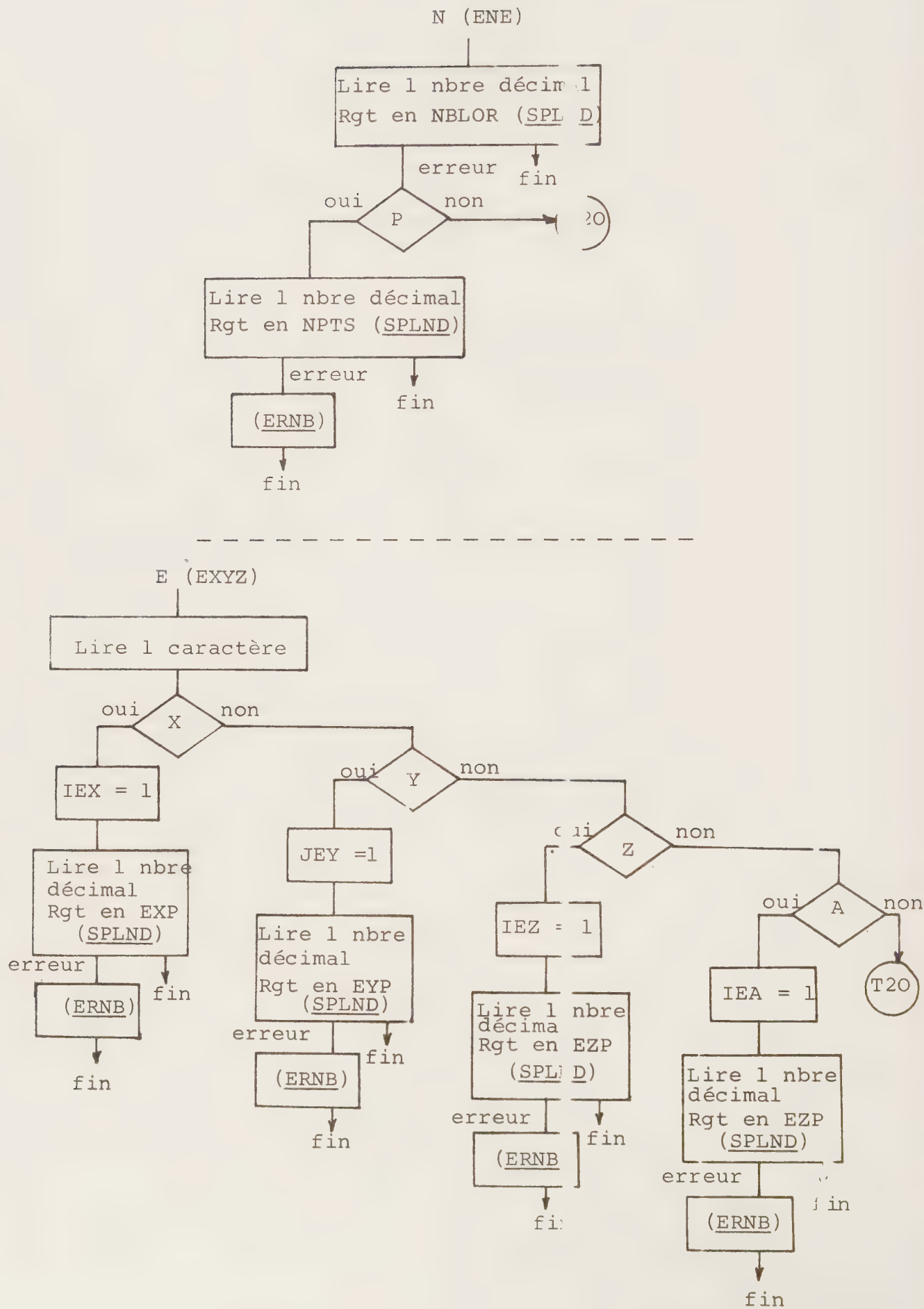
Le sous-programme TRBLO assure le tracé sur la table traçante de la figure définie par le bloc 2. Après un baisser de plume (BPLCO), la fonction relative au bloc 2 est testée. Si la variable FONCT2 = 0, il s'agit d'une droite et l'on se branche vers EXDRT qui l'exécutera. Sinon, il y a décodage de la fonction et branchement vers le sous-programme d'exécution correspondant.

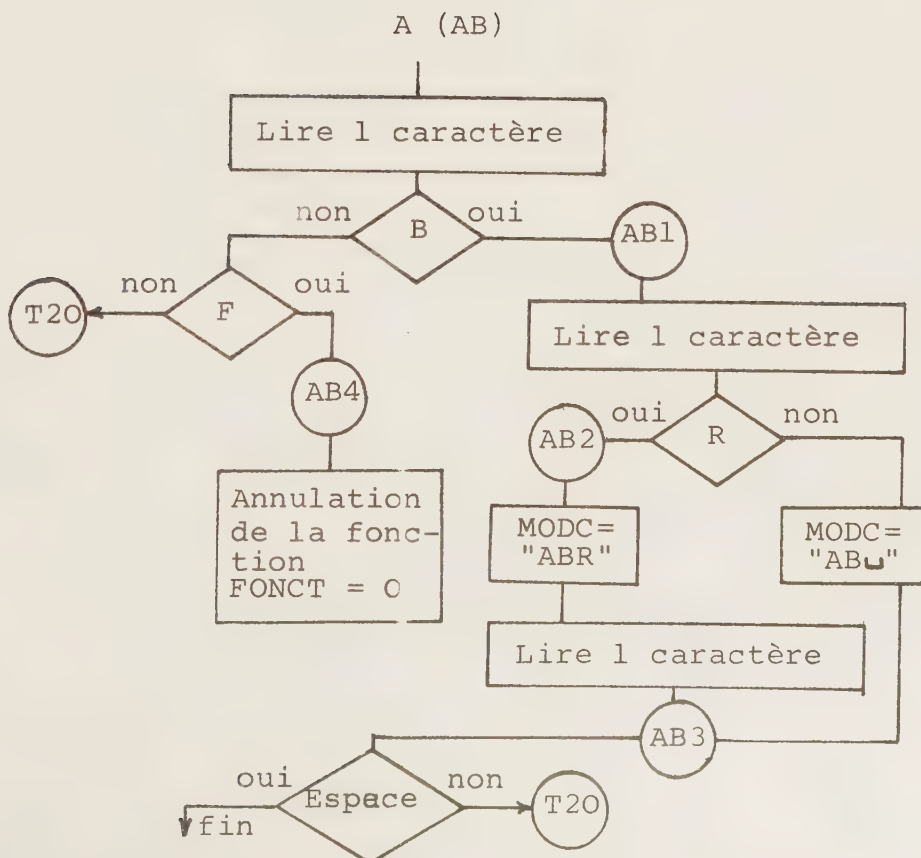
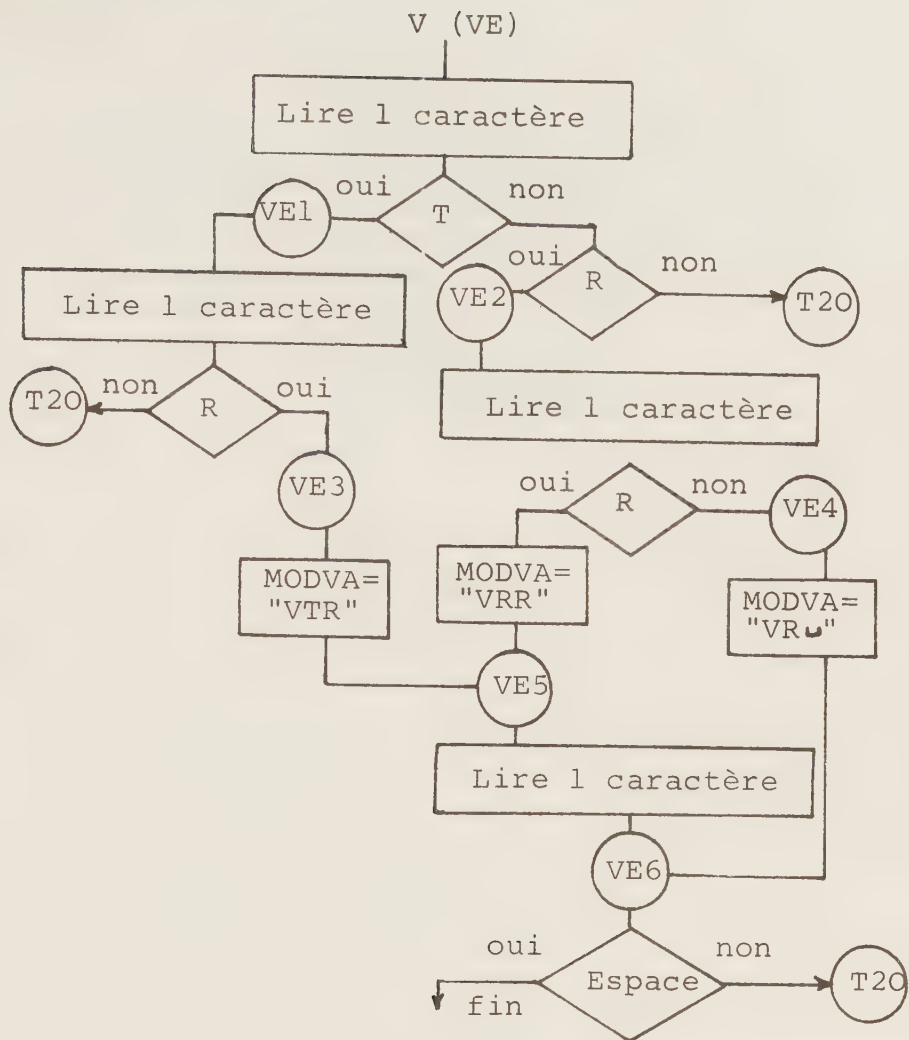




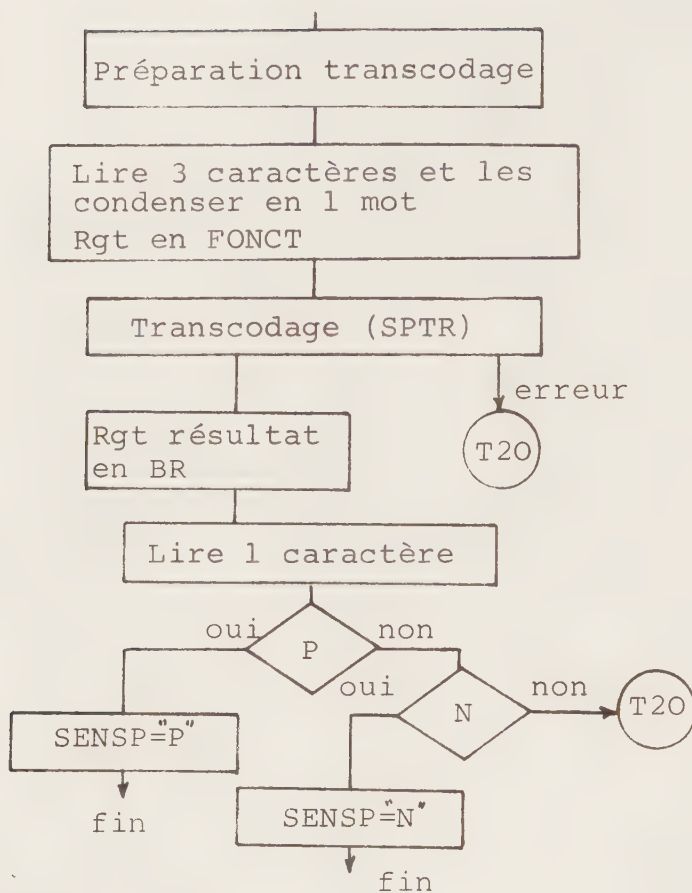


Cet organigramme est également valable pour les caractères :
 Y (YPO) - Z (ZPO) - I (IPO) - J (JPO) - K (KPO)

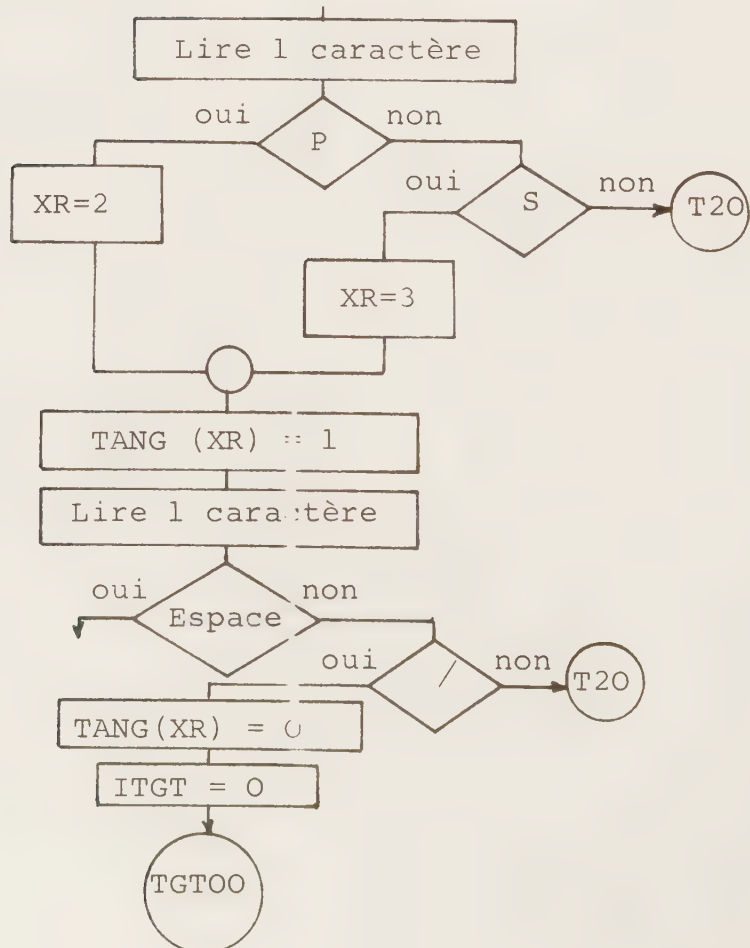


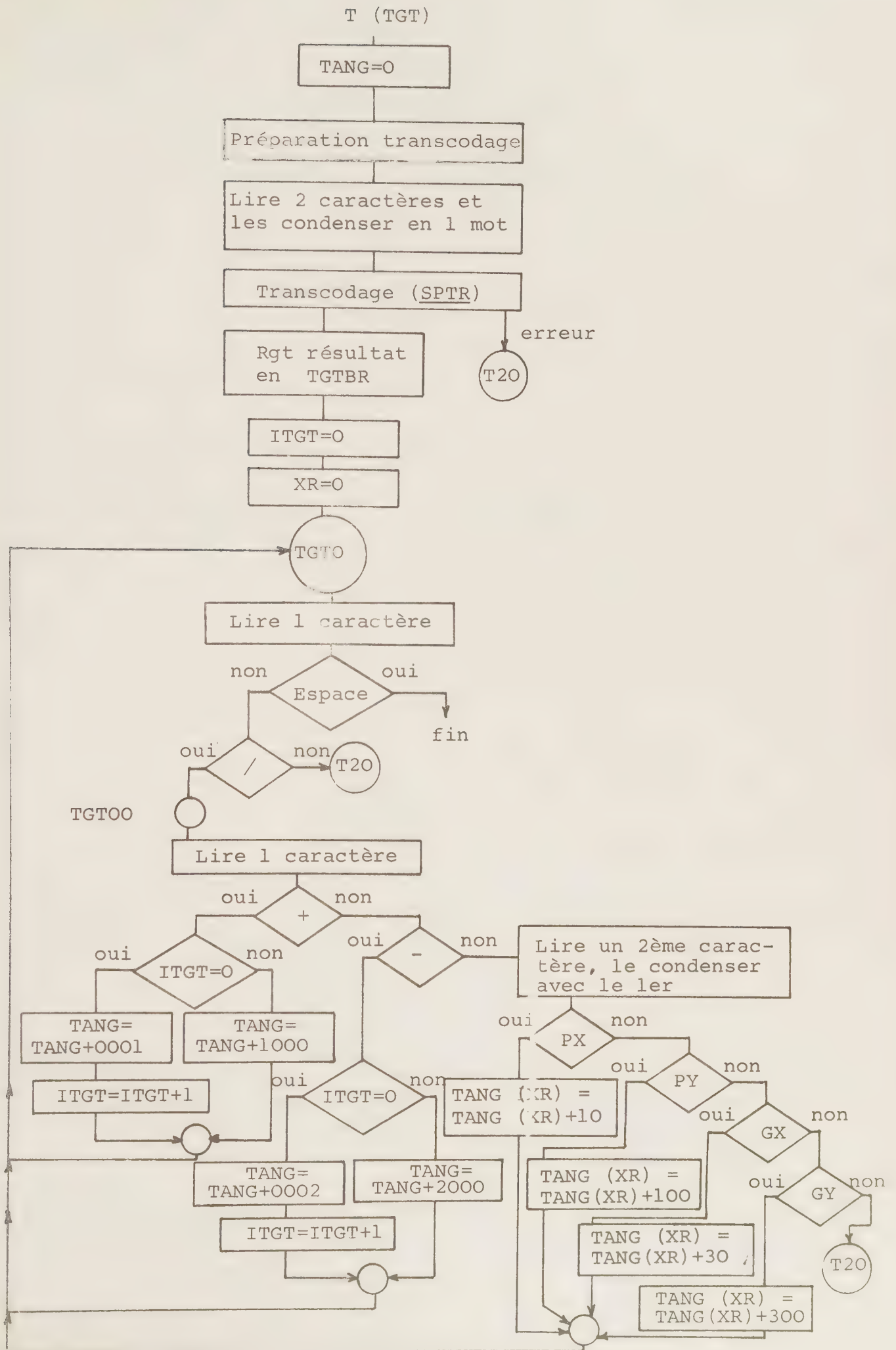


§ (DO)

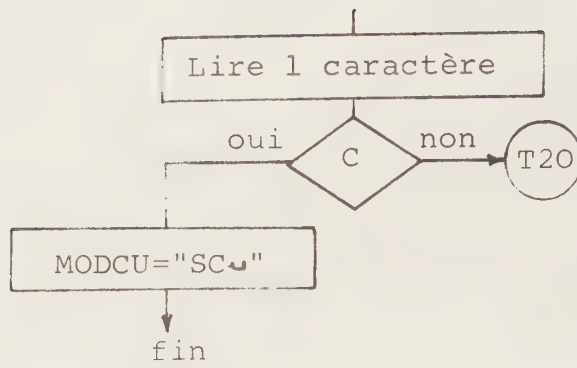


C (CEE)

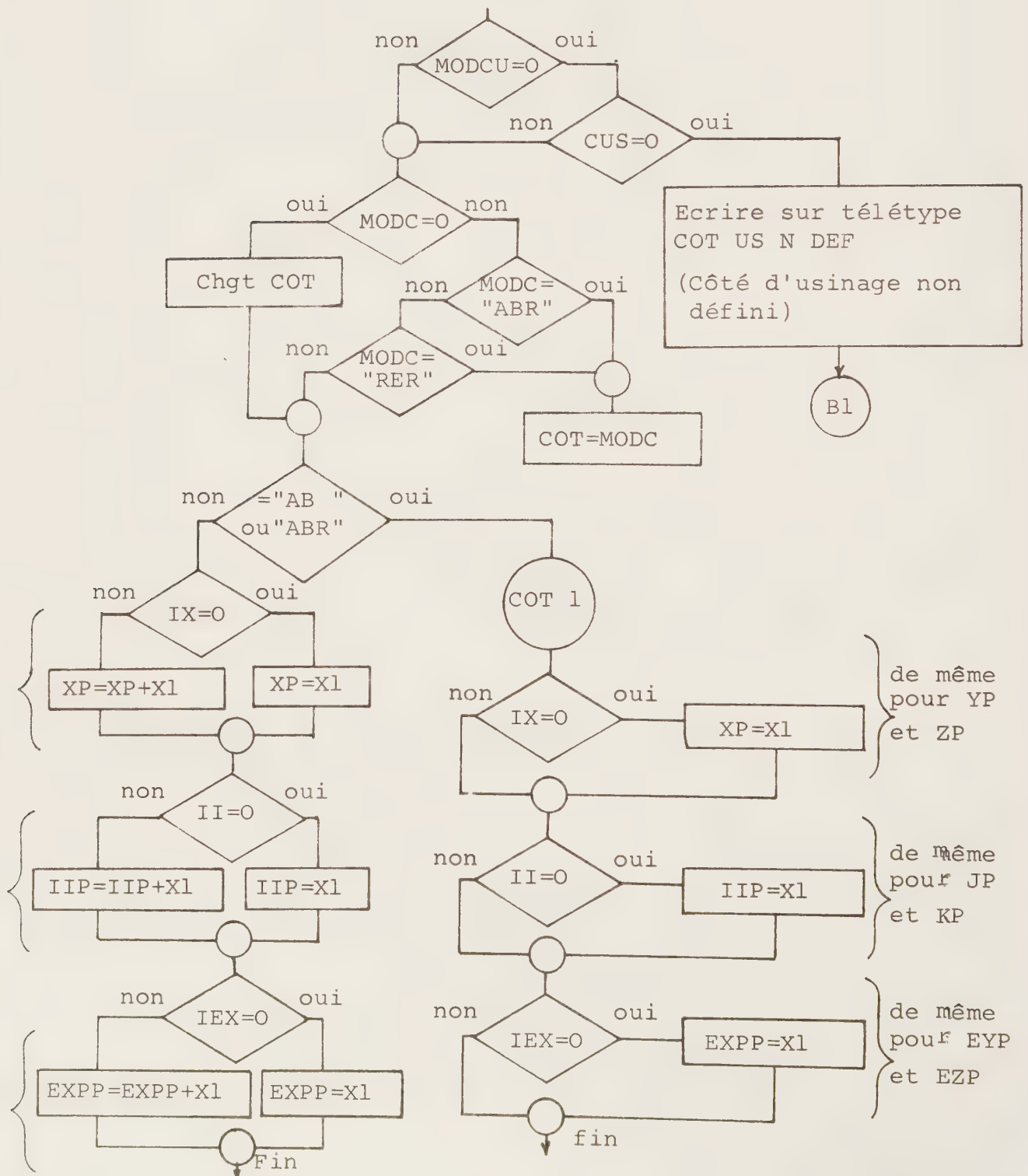


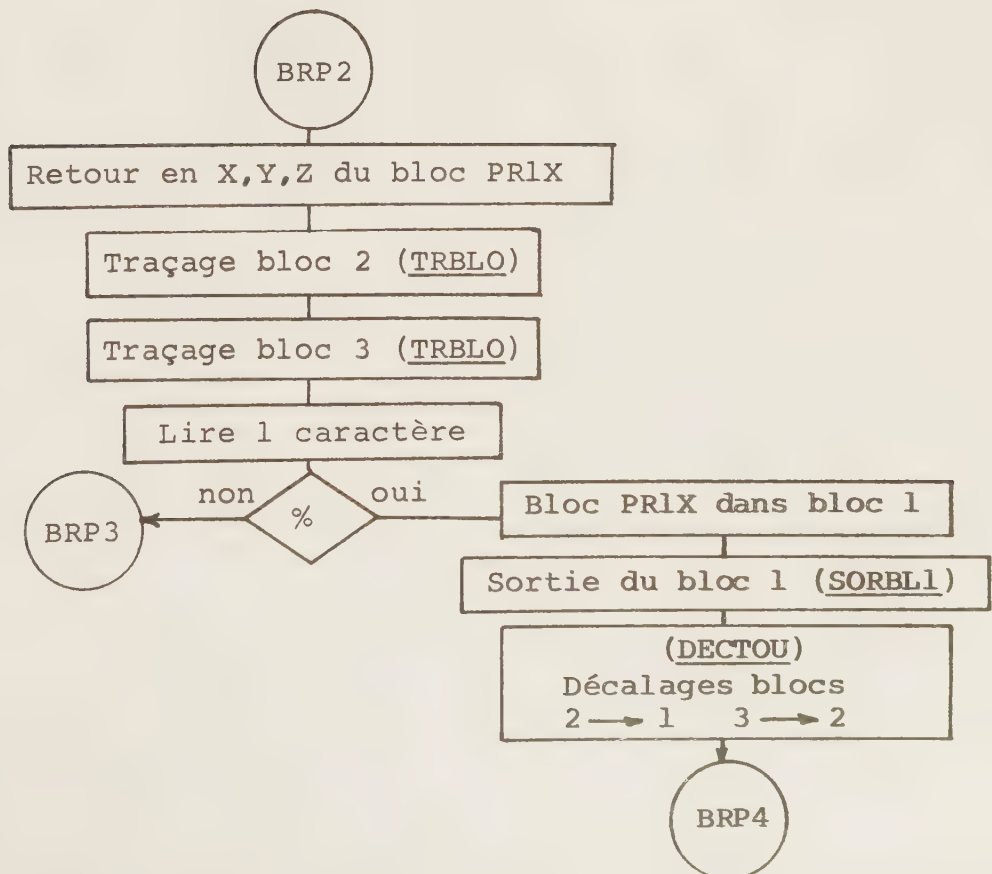
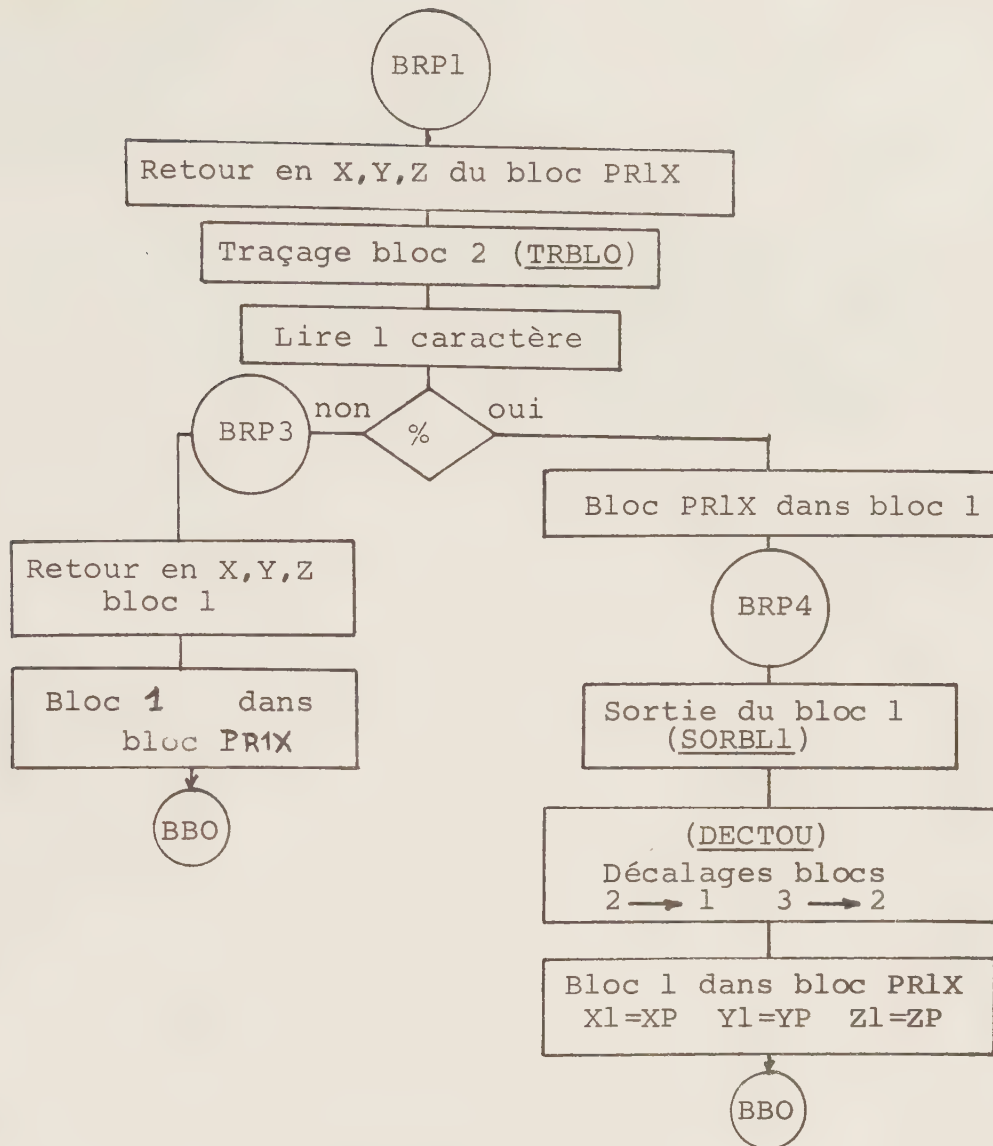


S (SCUS)

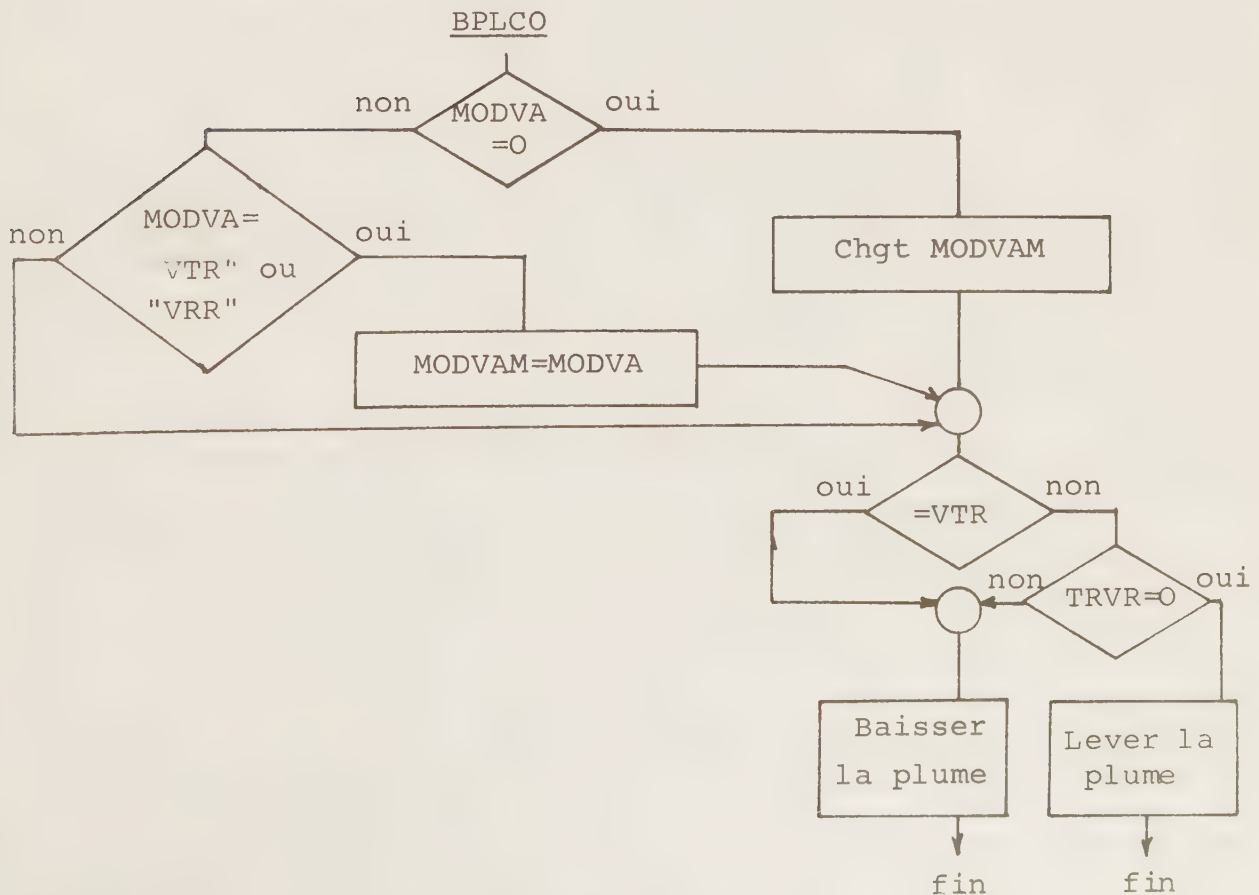
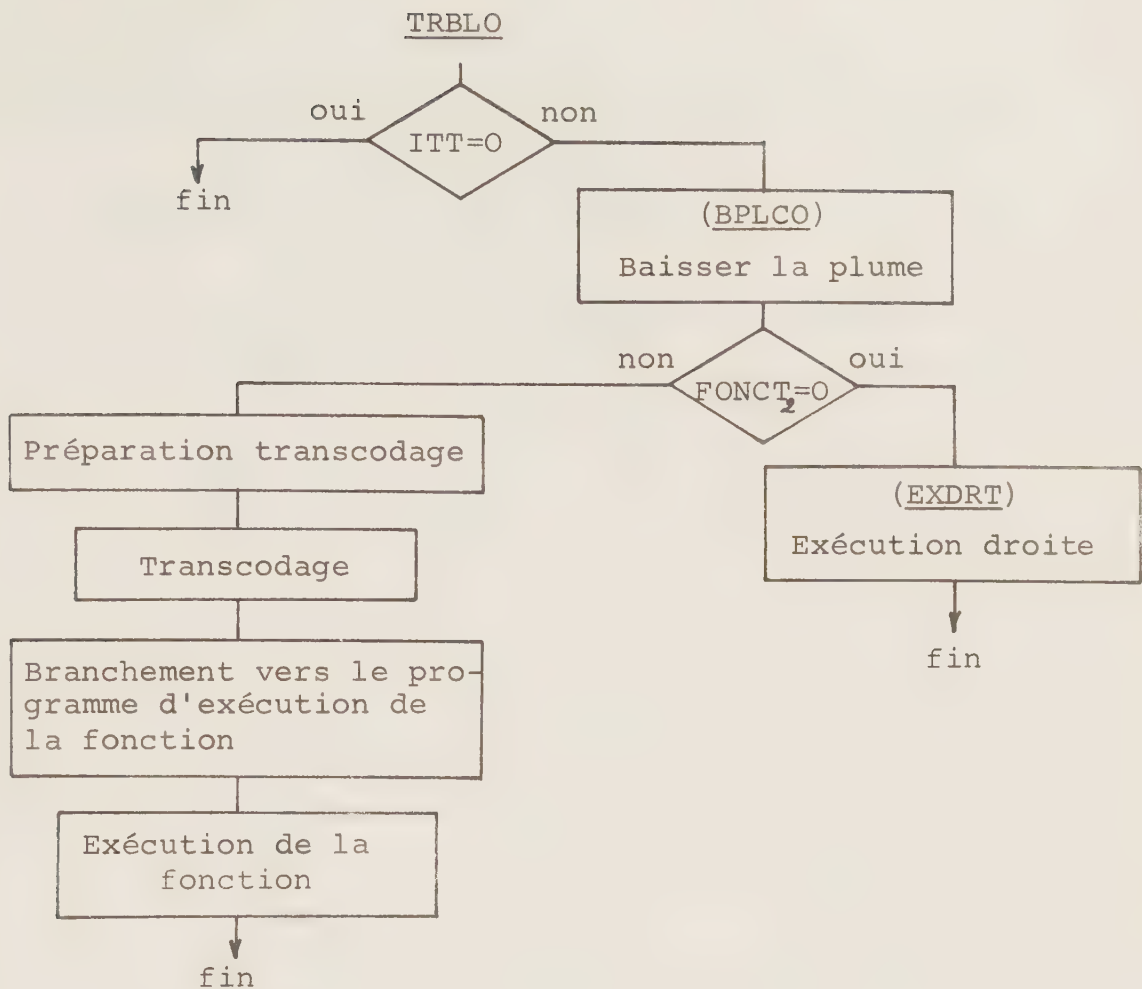


; (FB)





Le sous-programme BPLCO assure la descente de la plume après avoir examiné le mode de vitesse d'avance dans lequel on se trouve et la valeur de la variable TRVR.



V- Les sous-programmes d'interpolation et de calculs géométriques.

V-1 Sous-programme d'interpolation linéaire SPIL :

Les grandeurs d'entrée de ce sous-programme sont les coordonnées du point à atteindre, la vue et l'échelle du tracé.

Les variables XT et YT expriment les coordonnées de la plume à tout instant.

Après la sélection des coordonnées utiles en fonction de la vue choisie, le sous-programme SPIL examine quel est le plus grand déplacement à effectuer et son sens. Il initialise les variables MX et MY avec les mots correspondants aux incréments de base de la table traçante.

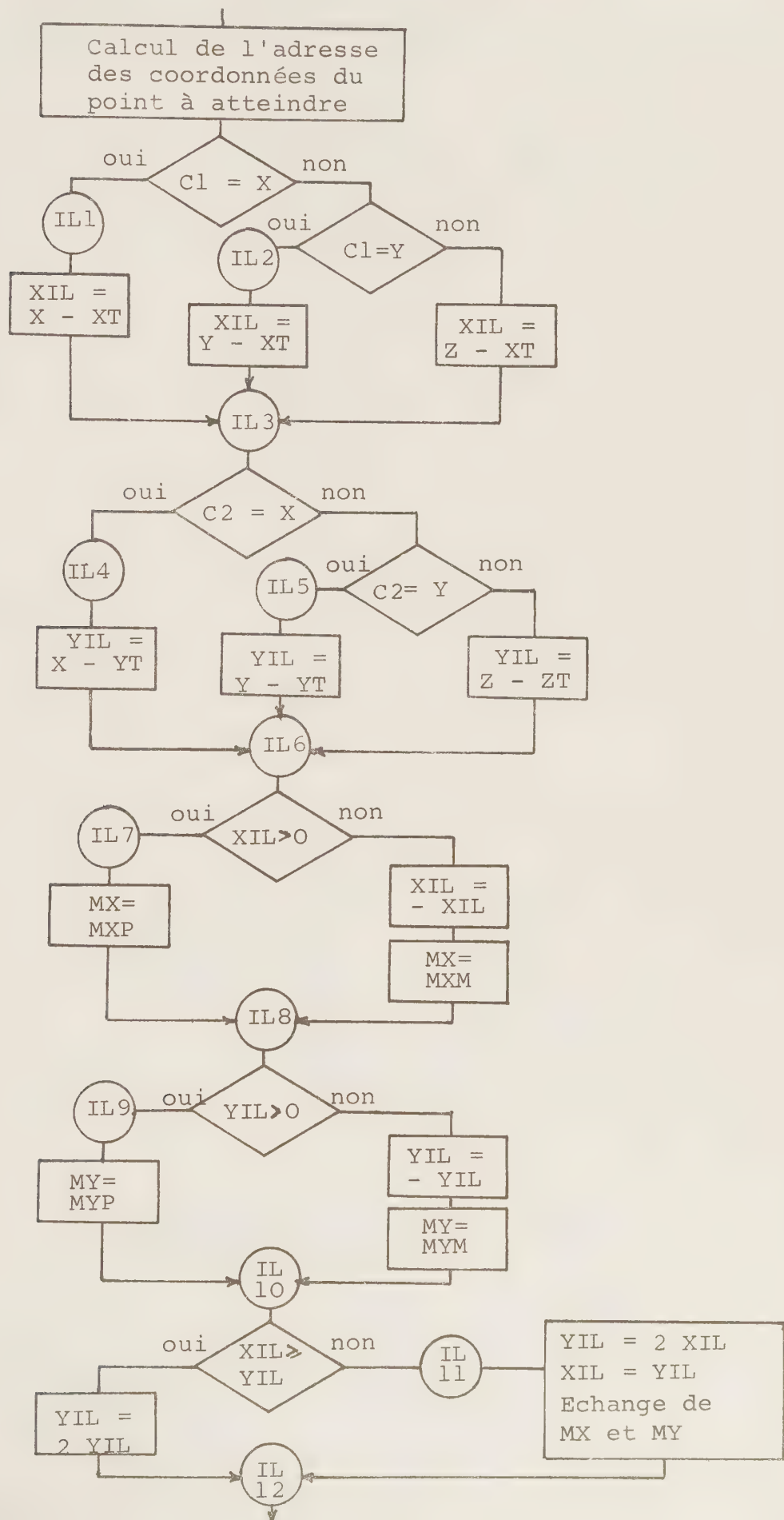
Pour l'interpolation on envoie un incrément sur l'axe correspondant au plus grand déplacement, si ce déplacement correspond à une variation supérieure à $1/2$ pas pour le deuxième axe, on lui envoie également un incrément.

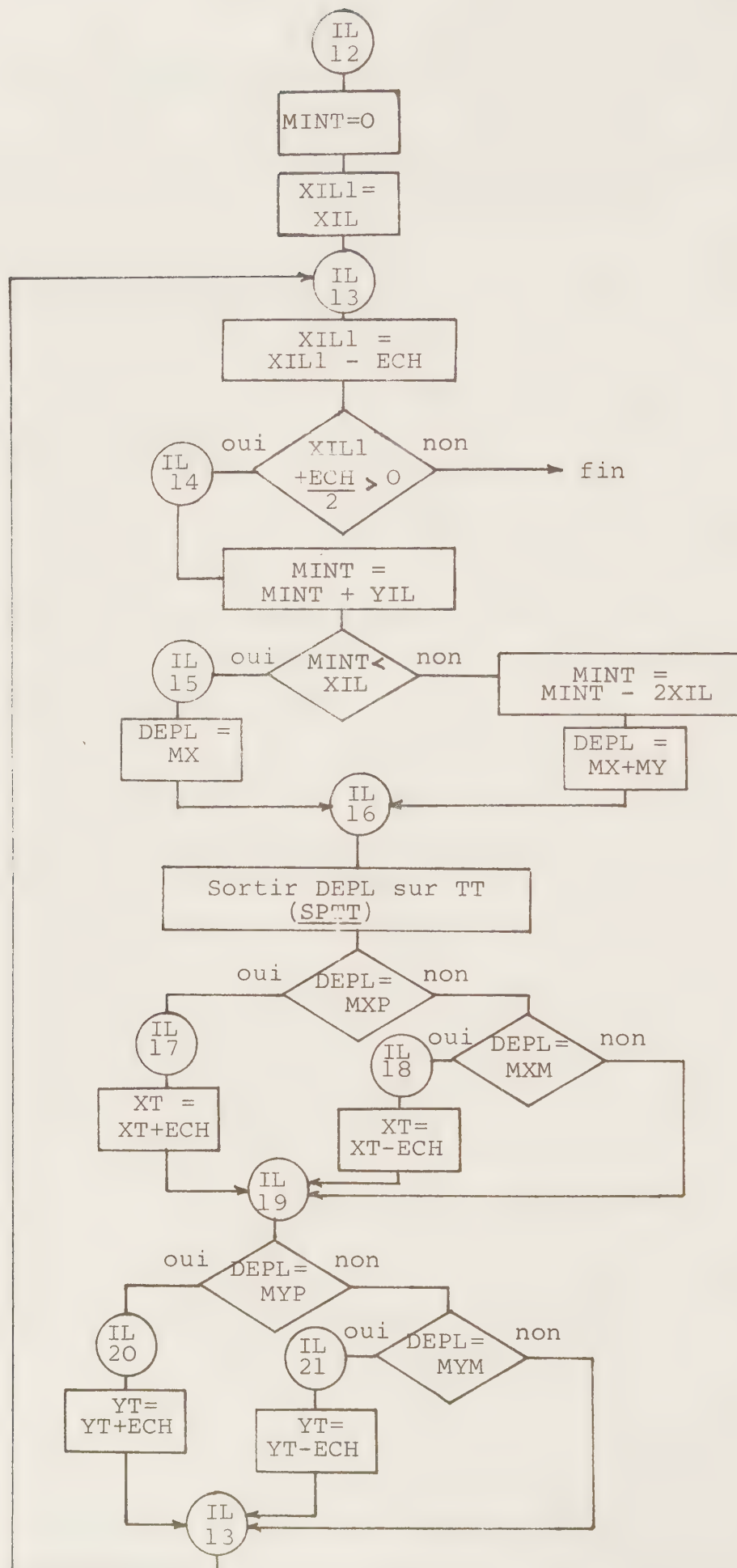
Après chaque période, on soustrait de la coordonnée correspondant à l'axe suivant lequel on s'est déplacé la valeur ECH qui exprime en $1/100$ ème la longueur d'un pas de table traçante compte-tenu de l'échelle choisie.

L'arrêt de l'interpolation est effectué quand, suivant l'axe du plus grand déplacement, l'écart entre la position de la plume et la position à atteindre est inférieur à $1/2$ pas.

SPIL

(Sous-programme d'interpolation linéaire)





V-2 Sous-programmes d'interpolation circulaire :

Plusieurs sous-programmes sont utilisés pour l'interpolation circulaire.

Le programme principal CALCER effectue les initialisations et possède deux sorties, l'une correspondant à la fin de l'interpolation, l'autre relative à un point quelconque du cercle interpolé.

Le sous-programme INTFIN initialise les variables IPX et IPY lors du calcul du premier point. Ces variables servant dans le sous-programme TESFIN à tester la fin de l'interpolation.

Le principe d'arrêt est le suivant :

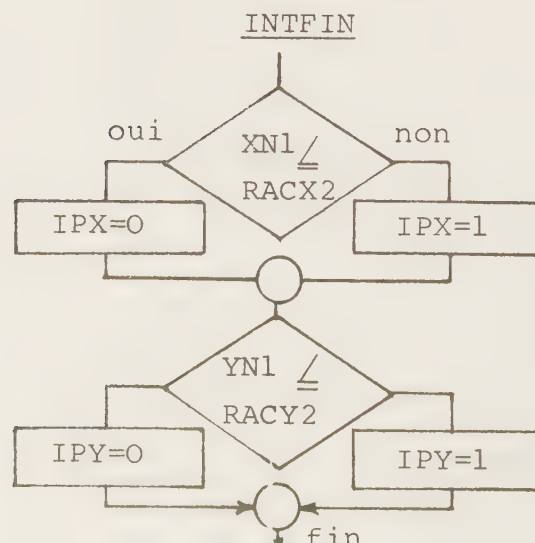
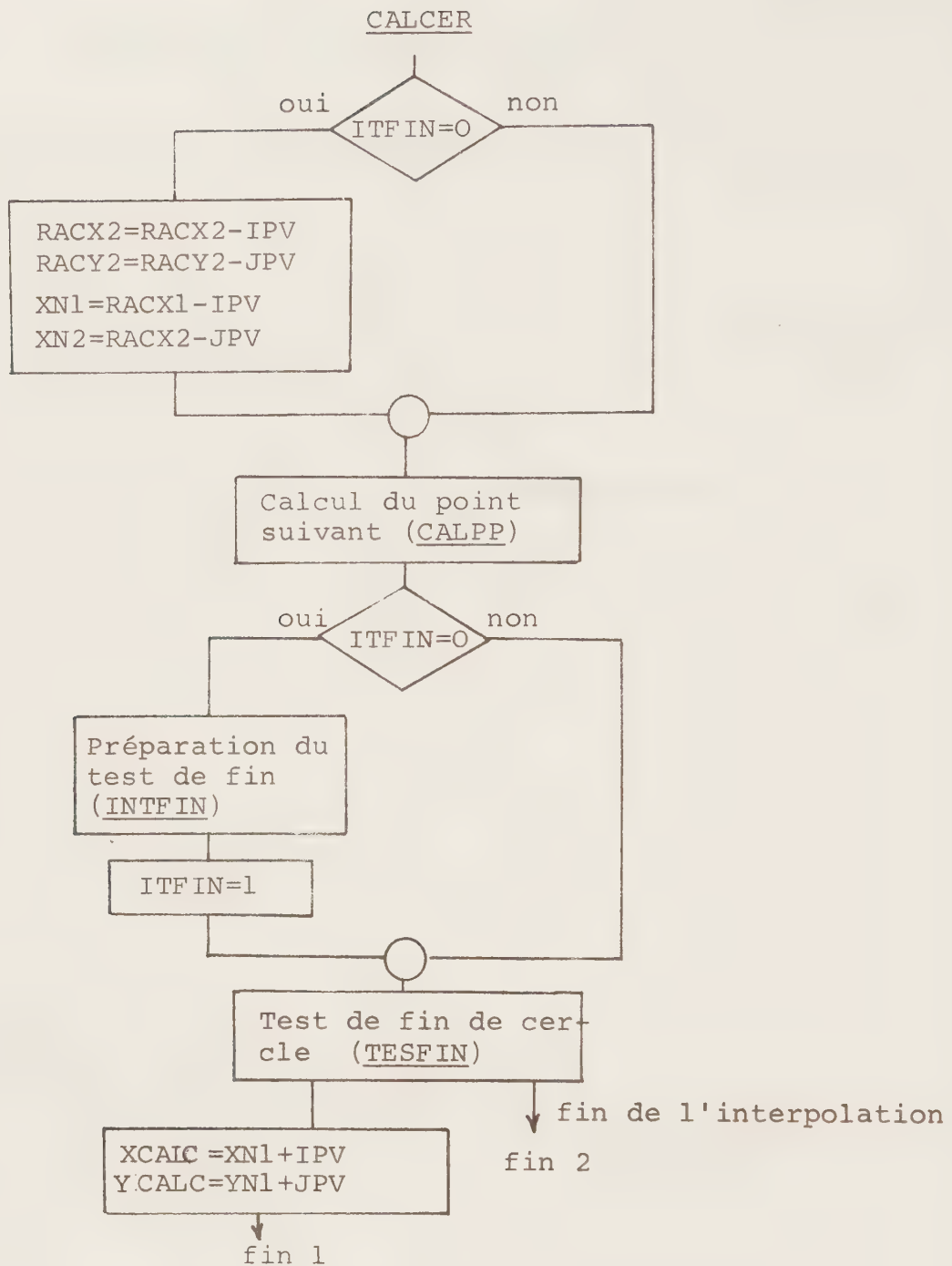
Au début de l'interpolation, la variable IPX est initialisée à 0 si le point de départ se trouve à gauche du point d'arrivée, et à 1 dans l'autre cas. La variable IPY est initialisée de la même façon en se référant à l'axe OY.

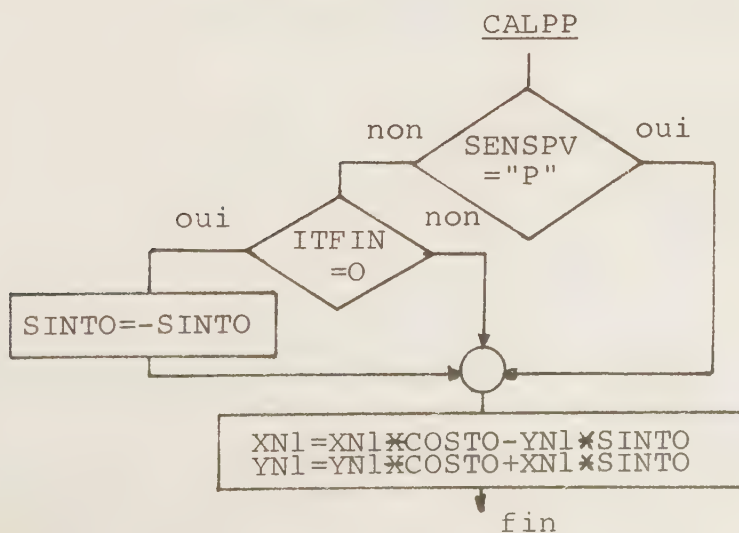
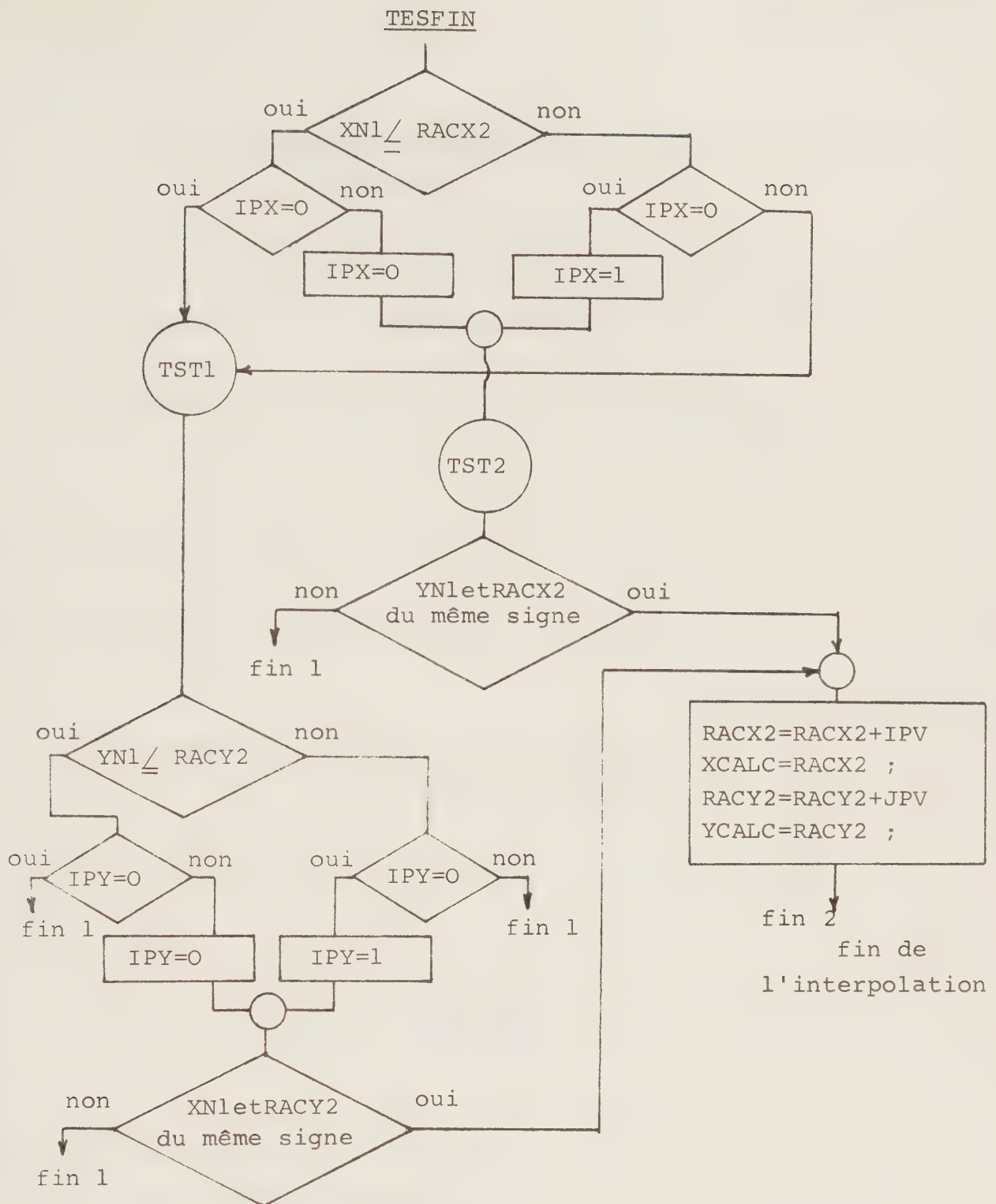
Après chaque période on compare la position du point calculé par rapport au point final et on évalue les grandeurs IPX et IPY. Si l'une des deux a été modifiée par rapport à la valeur précédente et si les coordonnées, suivant l'autre axe, des points calculé et final sont de même signe, on arrête l'interpolation.

Le sous-programme CALPP calcule à chaque période les valeurs des coordonnées du point suivant en utilisant la récurrence

$$\begin{aligned} x_{n+1} &= x_n \cos z - y_n \sin z \\ y_{n+1} &= y_n \cos z + x_n \sin z \end{aligned}$$

La variable SENSPV indiquant le sens de déplacement suivant le cercle, la valeur de SINTO sera inversée si SENSPV est différent de "P" (sens trigonométrique).





V-3 Sous-programmes de calcul géométriques :

Beaucoup de difficultés sont apparues pour l'écriture de ces sous-programmes du fait que nous nous sommes astreints à utiliser la représentation des grandeurs en virgule fixe.

En particulier, l'équation d'une droite est mémorisée comme suit :

- Si la droite admet une pente inférieure à 1 , l'équation sera

$$y = ax + b$$

- Si la droite admet une pente supérieure à 1 , l'équation sera

$$x = ay + b$$

Pour distinguer ces cas, une troisième variable (ECHXY) prend la valeur 0 ou 1 respectivement.

CERDRT : Calcul du point d'intersection d'une droite et d'un cercle.

Les équations :

$$Y = AX + B$$

$$X^2 + Y^2 = R^2$$

sont rapportées dans un système d'axes issus du centre, ce qui fournit

$$X = \frac{-AB \pm \sqrt{R^2(1+A^2) - B^2}}{1 + A^2}$$

Dans le cas où la droite est exprimée par $X = AY + B$, le résultat est identique mais relatif à la coordonnée Y.

Les résultats de ces calculs sont rangés en

$$\begin{vmatrix} \text{RACX1} \\ \text{RACY1} \end{vmatrix} \quad \text{et} \quad \begin{vmatrix} \text{RACX2} \\ \text{RACY2} \end{vmatrix} , \text{ le choix de la solution convenable étant}$$

effectué par le sous-programme CHOIX.

DRT DRT : Calcul du point d'intersection de 2 droites dont on connaît les équations.

Suivant la forme des équations des 2 droites, quatre cas se présentent :

$$\begin{array}{l} 1) \quad y = a_1 x + b_1 \\ \quad \quad y = a_2 x + b_2 \end{array} \quad x = \frac{b_2 - b_1}{a_1 - a_2}$$

$$\begin{array}{l} 2) \quad y = a_1 x + b_1 \\ \quad \quad x = a_2 y + b_2 \end{array} \quad x = \frac{a_2 b_1 + b_2}{1 - a_1 a_2}$$

$$\begin{array}{l} 3) \quad x = a_1 y + b_1 \\ \quad \quad y = a_2 x + b_2 \end{array} \quad y = \frac{a_2 b_1 + b_2}{1 - a_1 a_2}$$

$$\begin{array}{l} 4) \quad x = a_1 y + b_1 \\ \quad \quad x = a_2 y + b_2 \end{array} \quad y = \frac{b_2 - b_1}{a_1 - a_2}$$

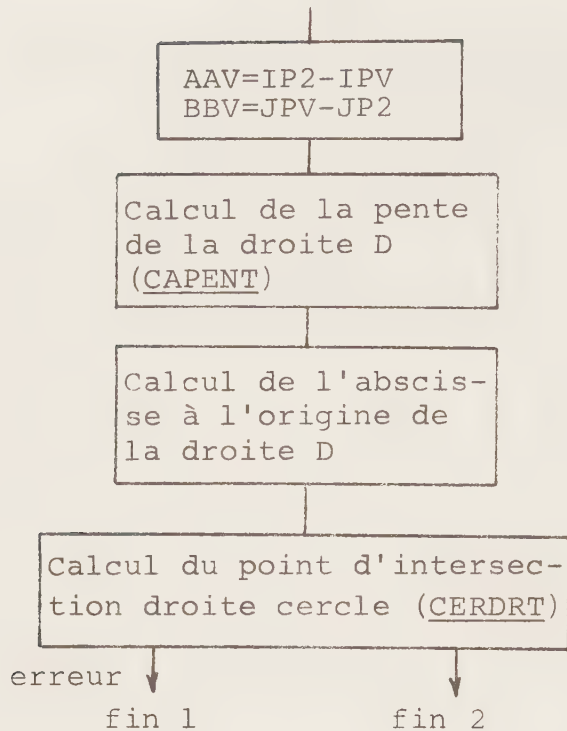
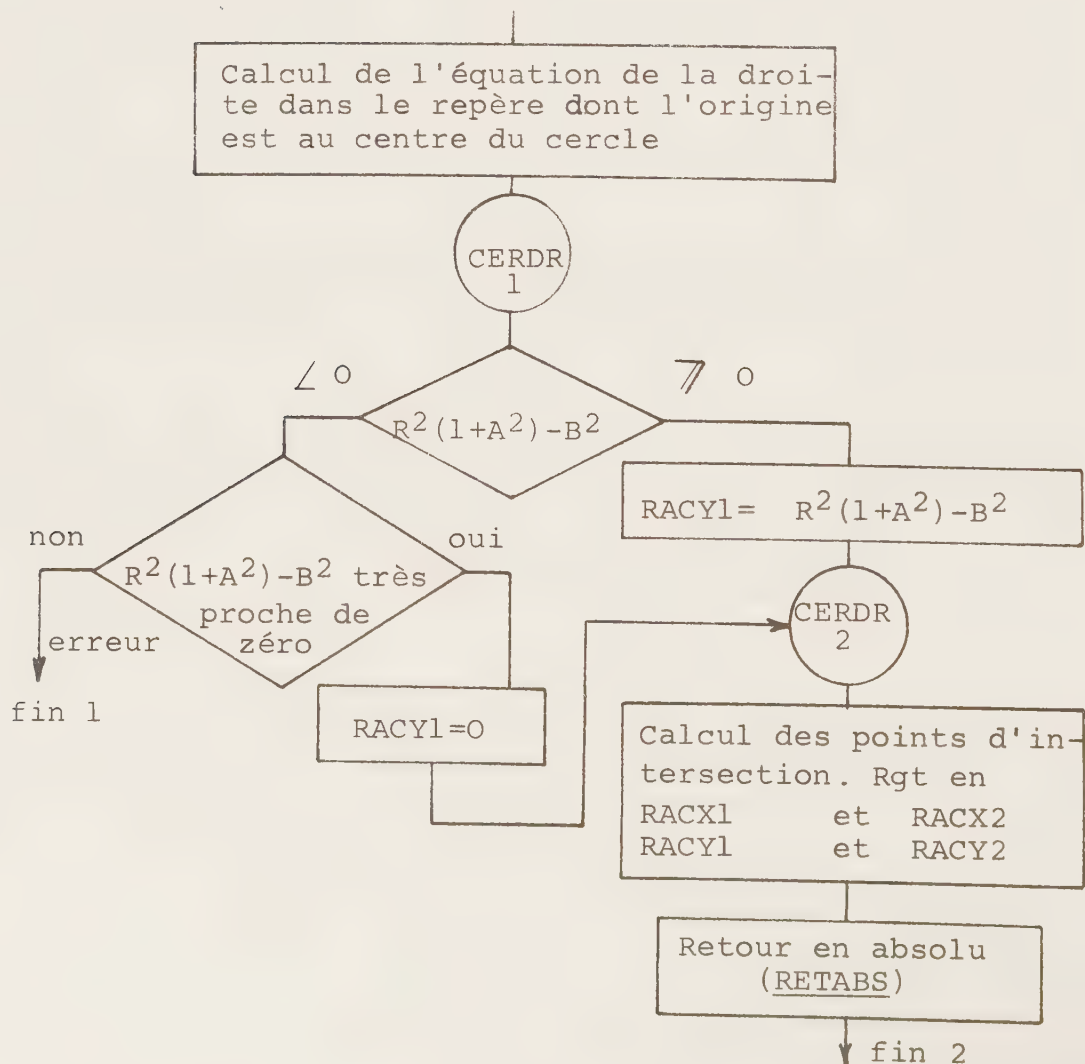
Les cas 1 et 4 admettent la même solution si l'on prend soin de permuter les coordonnées x et y. De même les cas 2 et 3.

Si la grandeur $1 - a_1 a_2$ ou $a_1 - a_2$, suivant le cas, est nulle le sous-programme DRTDRT renvoie au programme principal sans saut d'instructions, sinon on saute une place mémoire. La séquence d'appel est la suivante :

IRV DRTDRT ; (ou IR KRTDRT)

_____ ; retour si erreur

_____ ; retour normal

CERCERCERDRT

CAPENT : Calcul de la pente d'une droite connaissant les projections d'un segment sur les axes Ox et Oy.

Ce sous-programme détermine les valeurs des coefficients A , B et de la variable ECHXY qui précise la représentation utilisée

$$\text{ECHXY} = 0 \qquad y = Ax + B$$

$$\text{ECHXY} = 1 \qquad x = Ay + B$$

CALCT1 : Calcul du centre d'un cercle tangent à une figure connaissant le point de tangence et le rayon.

Après le calcul de la normale à la figure , on détermine la figure parallèle distante de la valeur du rayon de l'outil.

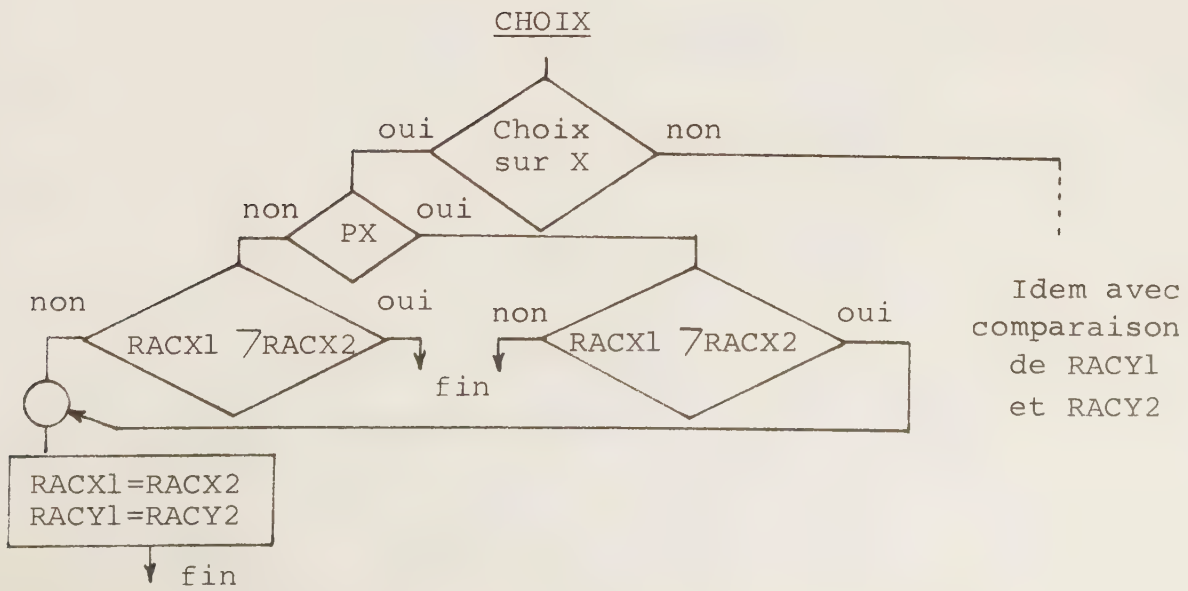
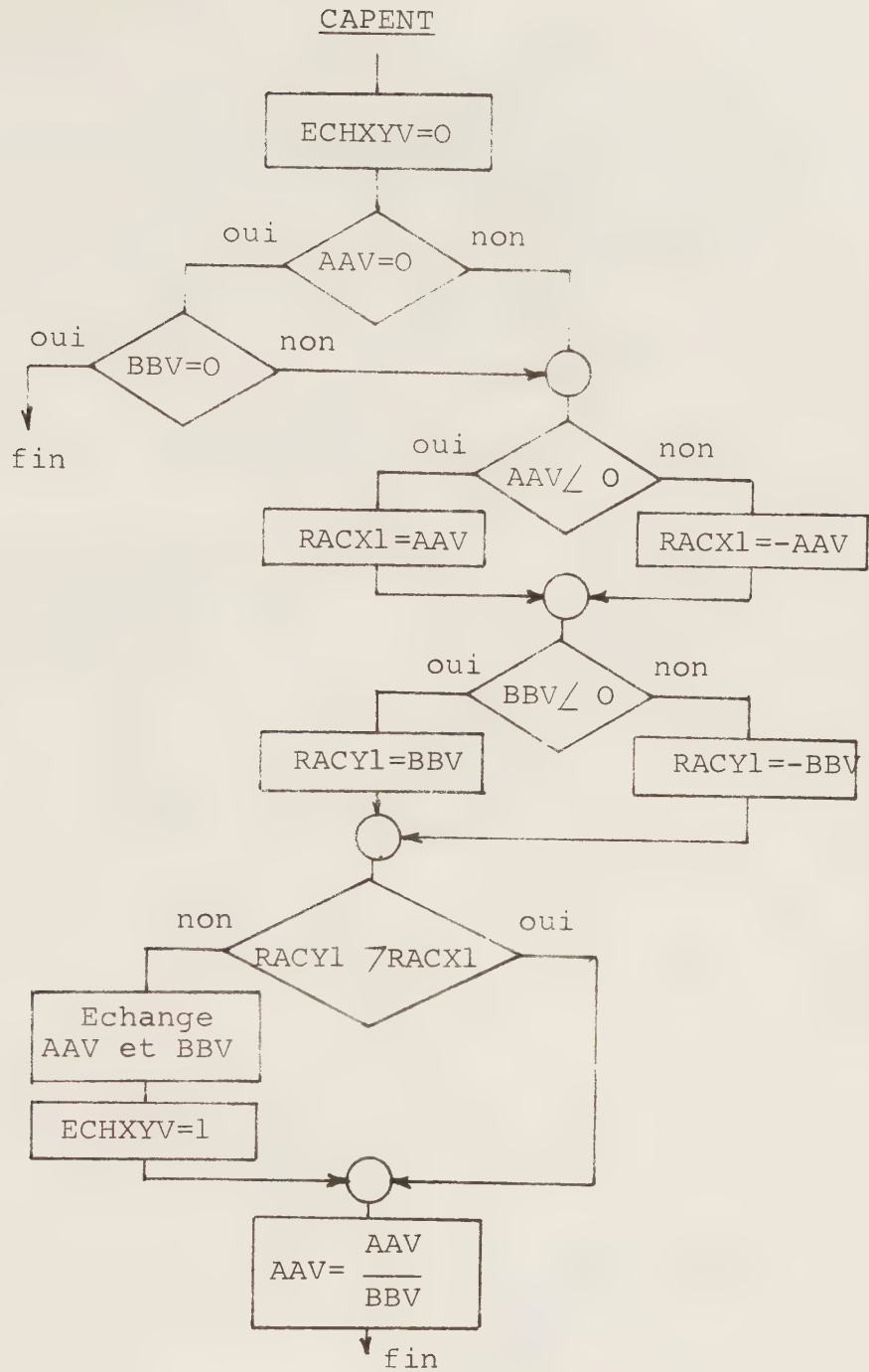
Le centre sera le point d'intersection de ces deux figures.

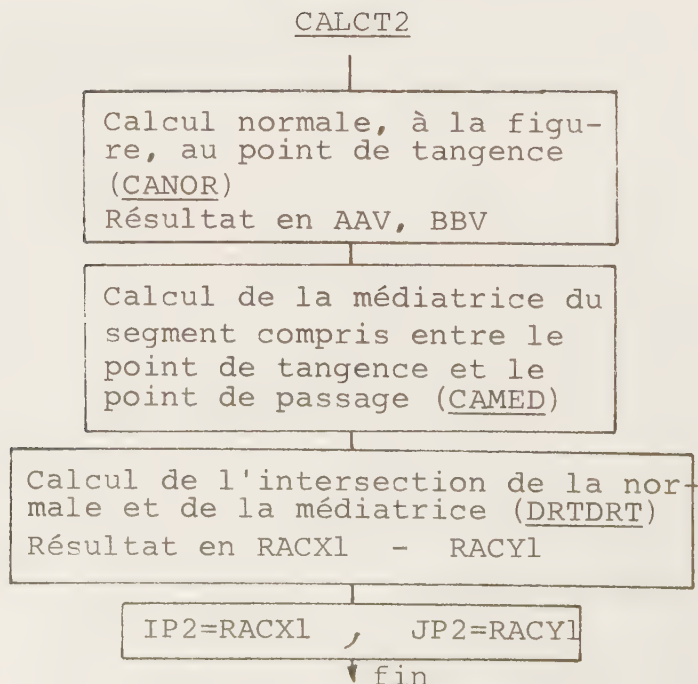
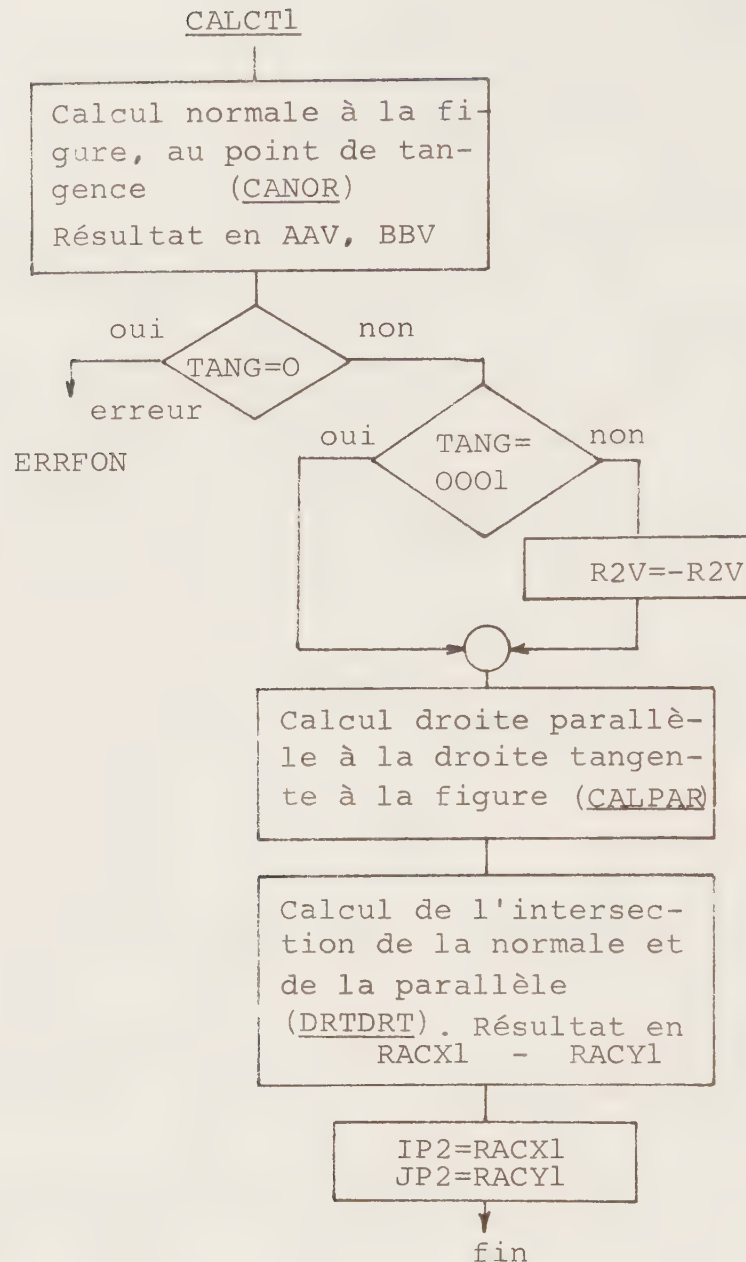
CALCT2 : Calcul du centre d'un cercle tangent à une figure, connaissant le point de tangence et un point de passage.

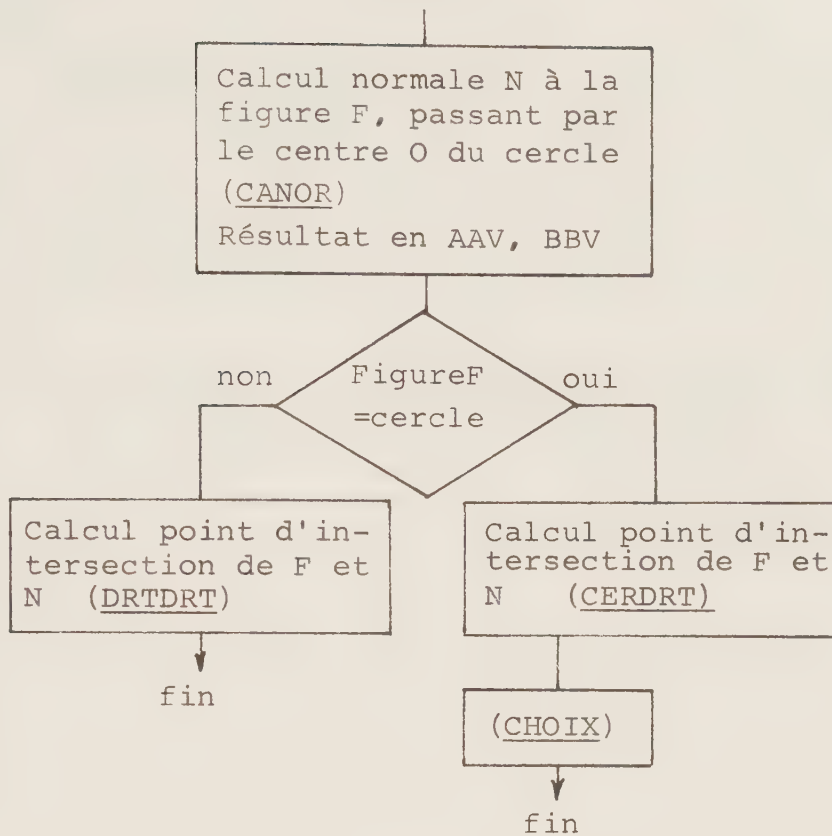
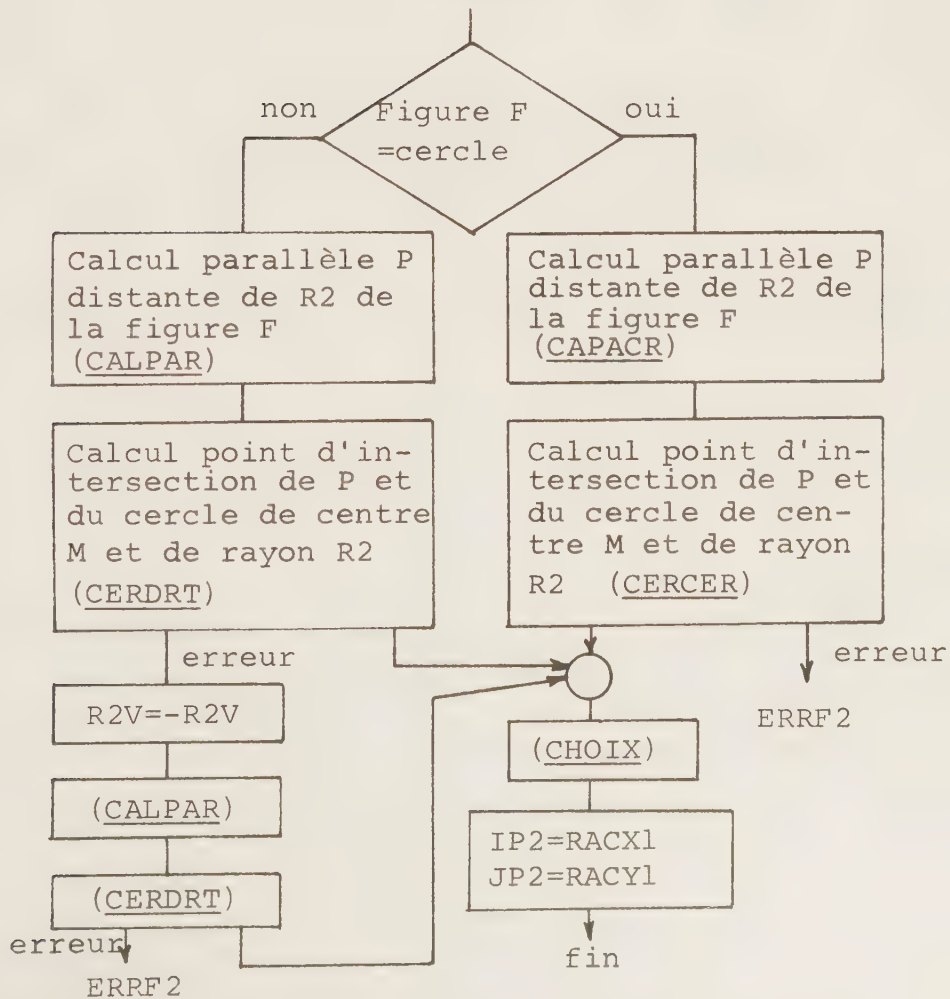
Après le calcul de la normale à la figure et de la médiatrice du segment compris entre le point de tangence et le point de passage, on détermine l'intersection de ces deux droites qui est le centre du cercle cherché.

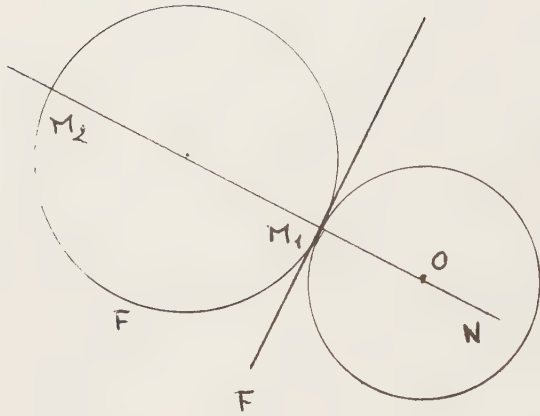
CALCT3 : Calcul du point de tangence à la figure F , connaissant le centre O du cercle.

Le point de tangence est l'intersection de la Figure et de sa normale passant par O.





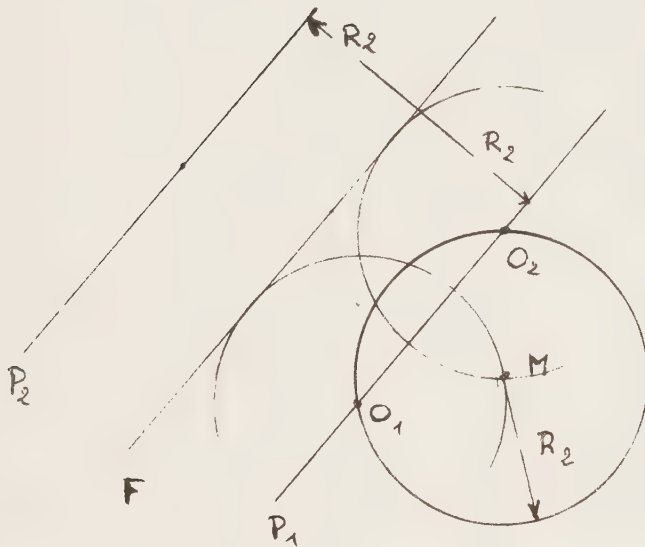
CALCT3CALCT4



- 1) Si F est une droite, il n'y a qu'une possibilité ,
- 2) Si F est un cercle, 2 solutions M_1 et M_2 se présentent et il faut choisir la solution convenable.

CALCT4 : Calcul des coordonnées du centre du cercle, connaissant son rayon R_2 et un point de passage M .

Le centre du cercle est l'intersection du cercle C de centre M et de rayon R_2 , et d'une parallèle à la figure F distante de R_2 .



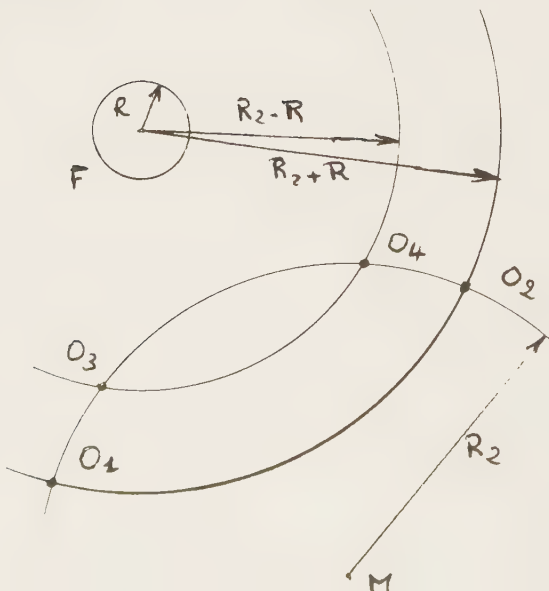
- 1) La figure F est une droite :

Il n'existe dans ce cas qu'une parallèle P qui convienne.

Le sous-programme CALCT4 calcule les équations des 2 parallèles P_1 et P_2 , de part et d'autre de la droite F et retiendra celle qui coupe le cercle C .

On conserve alors l'une des 2 solutions O_1 ou O_2 .

Si aucune parallèle ne coupe le cercle C , la fonction programmée est erronée.

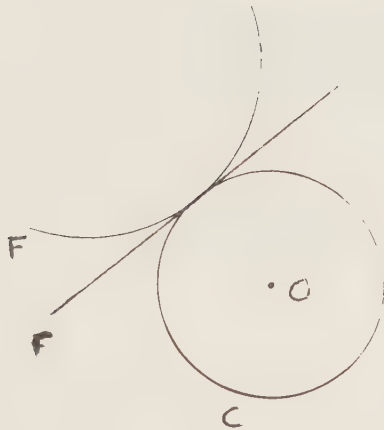


- 2) La figure F est un cercle :

Il existe dans certains cas 4 solutions possibles (voir figure).

Le sous-programme CAPACR permet d'isoler celui des cercles P_1 ou P_2 sur lequel se trouve le centre du cercle désiré. Le sous-programme CHOIX choisit alors la solution convenable.

CALCT5 : Calcul des coordonnées du point de tangence connaissant le centre et le rayon du cercle.



Ce point de tangence est le point de contact entre le cercle C et la figure F.

C'est pour ce sous-programme que nous avons toléré dans les programmes, CERDRT et CERCER, un discriminant légèrement négatif que nous ramenons à zéro. En effet, il se pourrait sans cela que l'on ne trouve pas de point de contact, à cause des erreurs de chute qui peuvent apparaître dans le calcul du centre, ou du rayon du cercle.

CALPAR : Calcul de l'équation de la parallèle à une droite, distante de $R2V$.

Si l'équation de la droite est donnée par $y = ax + b$, l'équation de la parallèle est $y = ax + b + R2V \sqrt{1 + a^2}$.

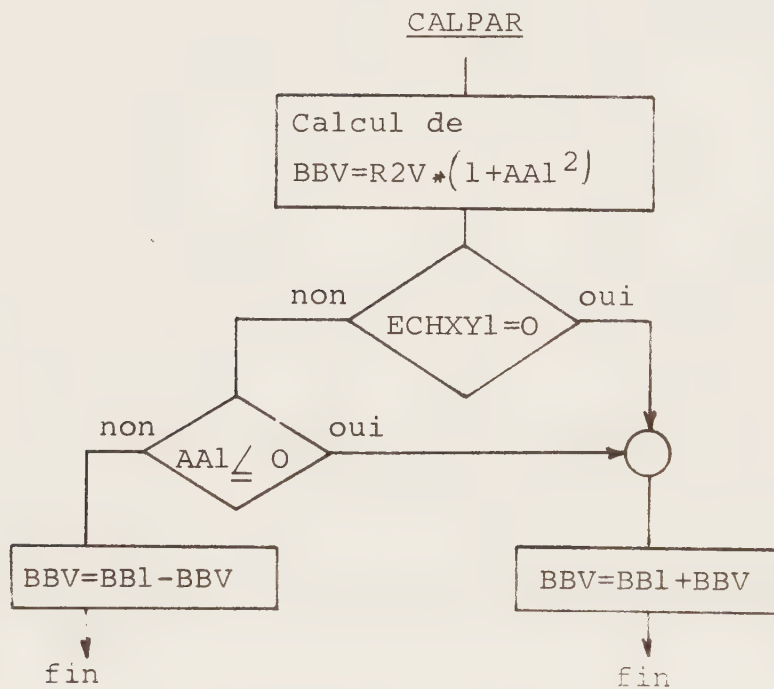
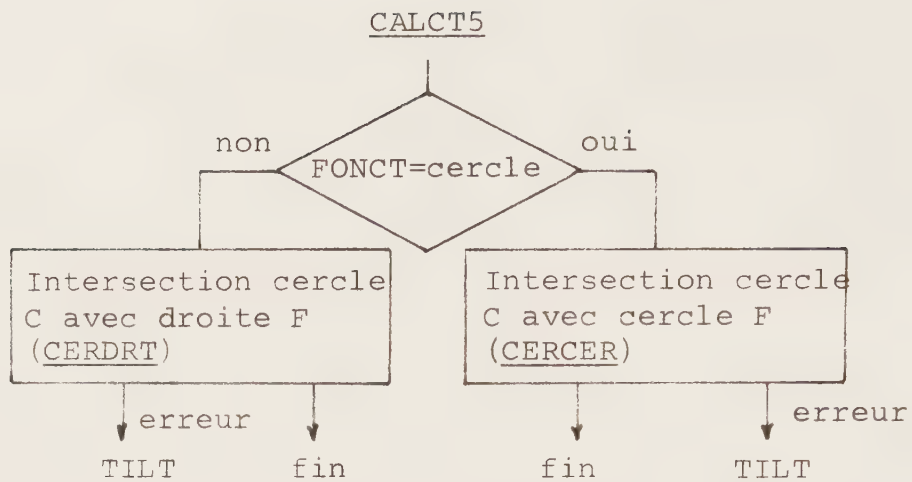
Si l'équation de la droite est donnée par $x = ay + b$, on rencontre 2 cas

- a = 0 : l'équation de la parallèle est $x = ay + b + R2V \sqrt{1 + a^2}$
- a ≠ 0 : on a $x = ay + b - R2V \sqrt{1 + a^2}$

CANOR : Calcul de l'équation de la normale passant par M, à une figure F.

1) Si F est une droite d'équation $y = ax + b$ la normale a pour équation

$$x = -ay + b'$$



Si F est une droite d'équation $x = ay + b$ la normale a pour équation

$$y = -ax + b'$$

La grandeur b' est déterminée par le sous-programme CAABSO.

2) Si F est un cercle, on détermine la pente de la droite passant par M et O centre du cercle. Puis l'abscisse à l'origine est calculée par CAABSO.

CAABSO : Calcul de l'abscisse à l'origine d'une droite dont on connaît la pente et un point de passage de coordonnées x_1, y_1 .

Si la droite a pour équation $y = ax + b$, $b = y_1 - a_{x1}$;

Si la droite a pour équation $x = ay + b$, $b = x_1 - a_{y1}$.

CAMED : Calcul de l'équation de la médiatrice du segment S dont on connaît l'origine et l'extrémité.

Après le calcul de la pente de la normale au segment, on détermine les coordonnées de son milieu, puis l'abscisse à l'origine de la médiatrice.

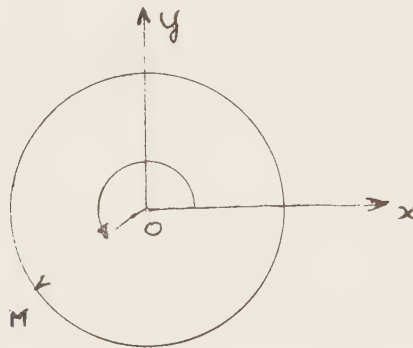
CACENT : Calcul des coordonnées du centre du cercle tangent aux figures précédente FP, et suivante FS.

Ce centre se trouve à l'intersection des parallèles aux figures FP et FS, distantes de $R2V$ de celles-ci.

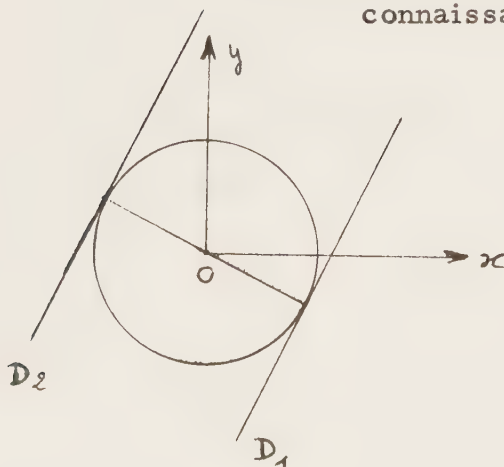
Le choix des parallèles est effectué lors de la programmation de la fonction.

Si les parallèles admettent 2 points d'intersection, le choix de la solution se fait à l'aide du sous-programme CHOIX.

CAFINC : Calcul des coordonnées du point final M d'un cercle C, connaissant le centre et le rayon du cercle, et l'angle $(\vec{Ox}, \vec{OM})$



TGTC1 : Calcul de l'équation de la droite D tangente à un cercle C, connaissant la pente de la droite.



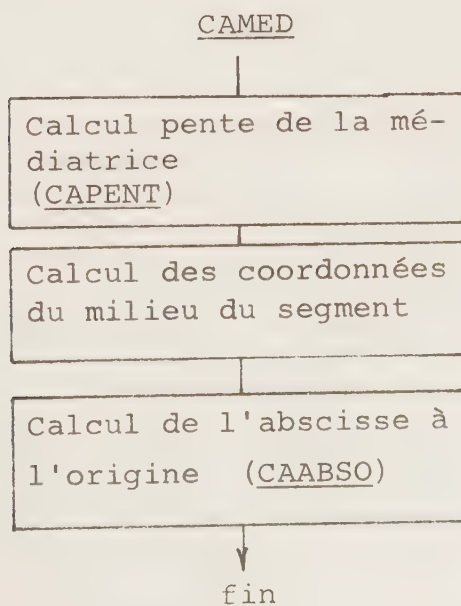
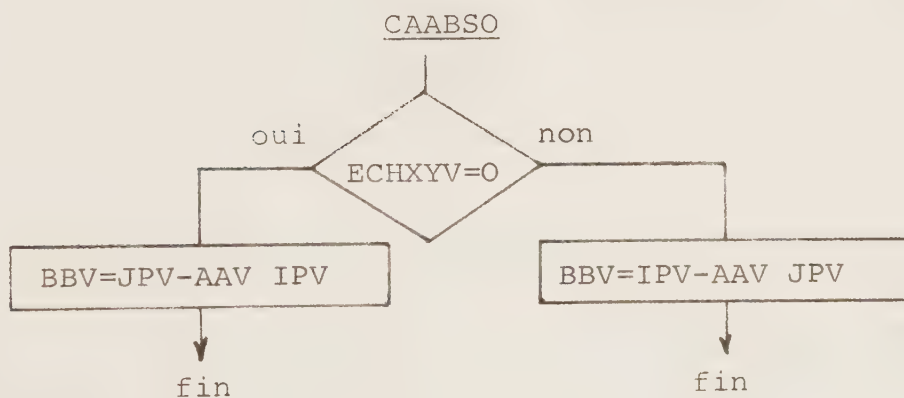
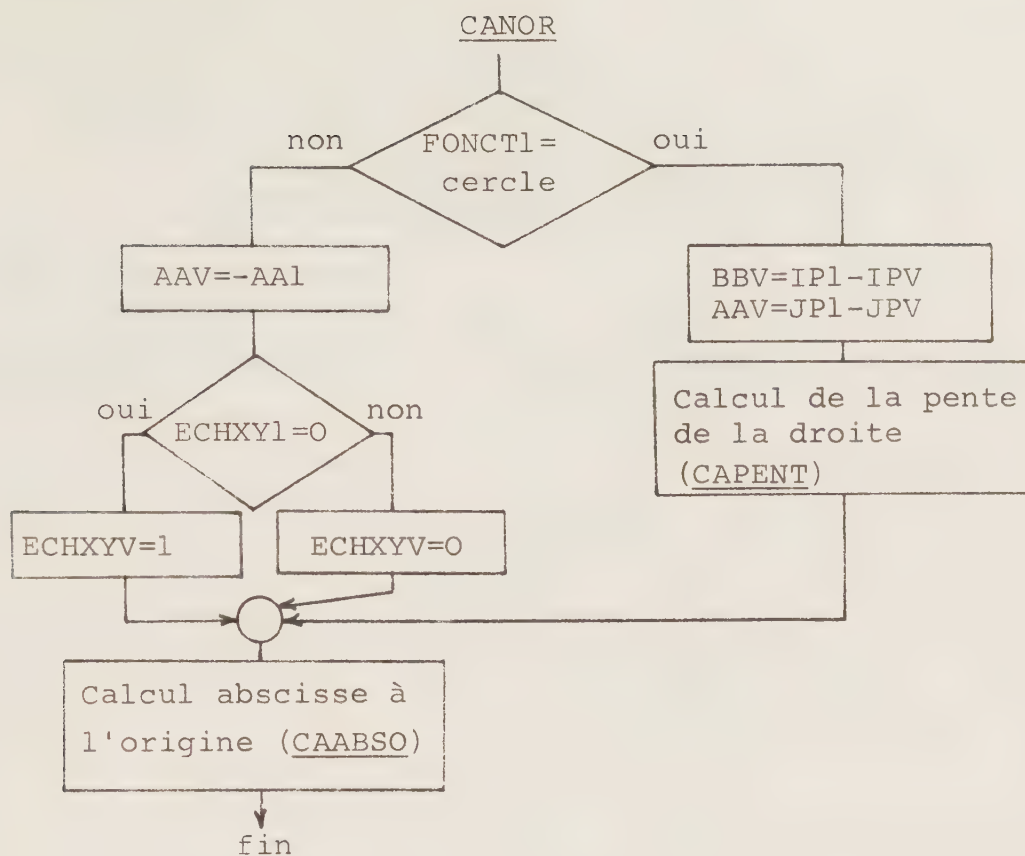
Calcul des coordonnées du point de tangence

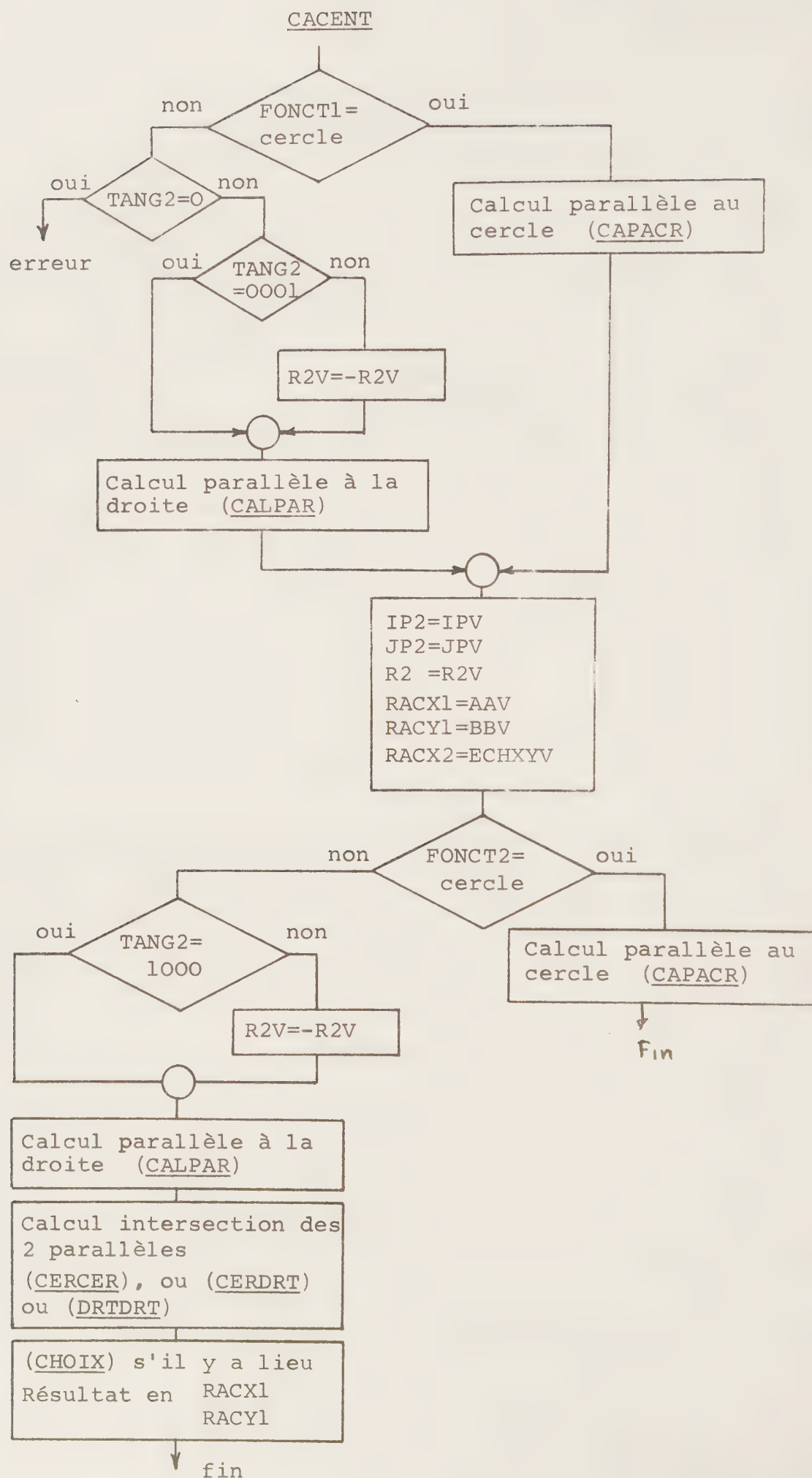
On calcule l'équation de la normale à la droite D passant par O.

Le choix entre les 2 points de tangence possible est fait par le sous-programme CHOIX.

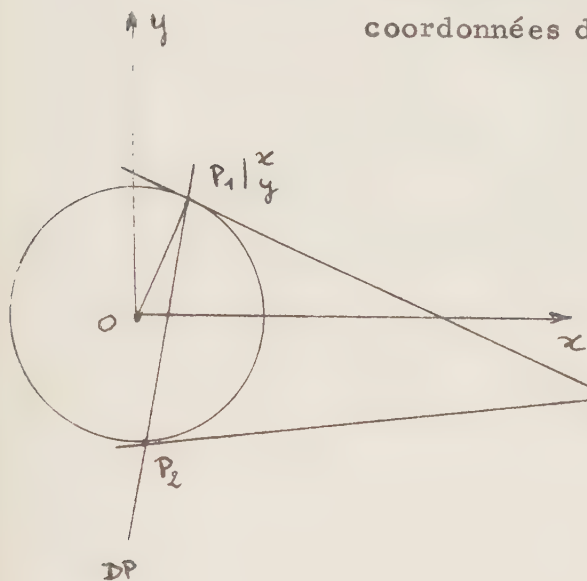
TGTC2 : Calcul de l'équation de la droite D tangente à un cercle C, connaissant le point de tangence.

On calcule la pente de la droite, puis son abscisse à l'origine.





TGTC3 : Calcul de l'équation de la droite D tangente à un cercle C, connaissant un point de passage M de la droite. Calcul des coordonnées du point de tangence.



Dans les axes Ox , Oy, on a :

- Equation du cercle

$$1) x^2 + y^2 = R^2$$

Produit scalaire $\vec{OP_1} \cdot \vec{P_1M} = 0$

$$x(x-x_1) + y(y-y_1) = 0$$

$$2) x^2 - xx_1 + y^2 - yy_1 = 0$$

$$1)-2) xx_1 + yy_1 = R^2$$

C'est l'équation de la droite DP.

Il suffit donc de calculer ainsi l'équation de DP, puis les intersections de cette droite avec le cercle C et enfin de choisir la solution qui nous convient.

TRBL3 : Sous-programme de traitement du bloc 3 .

Lorsque la fonction programmée nécessite la connaissance du bloc suivant, il faudra traiter les informations dès leur introduction dans le bloc 3.

Il est évident que celles-ci ne doivent pas faire appel à un nouveau bloc.

V-4 Sous-programmes d'exécution des différentes figures :

Les organigrammes des programmes EXDRT, EXCER, EXPCE, EXPDR et DEPEXC sont assez explicites. Ces sous-programmes sont appelés par le sous-programme TRBLO (tracé d'un bloc d'informations).

TGTC1

Calcul pente de la normale à la droite

BBV=0

Calcul intersection de cette normale avec le cercle C (CERDRT)

Résultat en RACX1
RACY1

(CHOIX)

Calcul de l'abscisse à l'origine de la tangente (CLABSO)

fin

TTTC3

Calcul pente de la droite DP (CAPENT)
Résultat en AAV

$$BBV = \frac{(R2V)^2}{BBV}$$

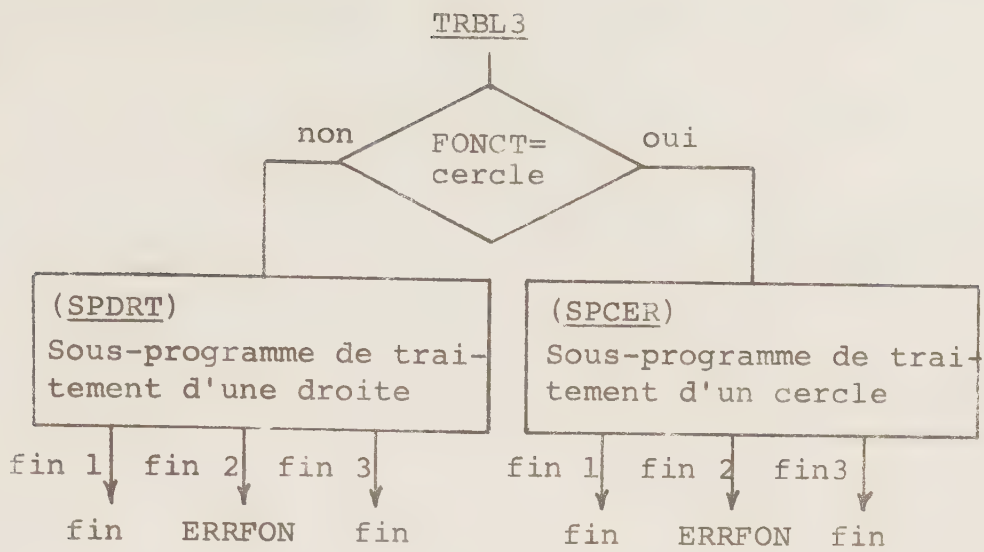
non ECHXYV=0 oui

BBV=-BBV

Intersection cercle C et droite DP (CERDRT)

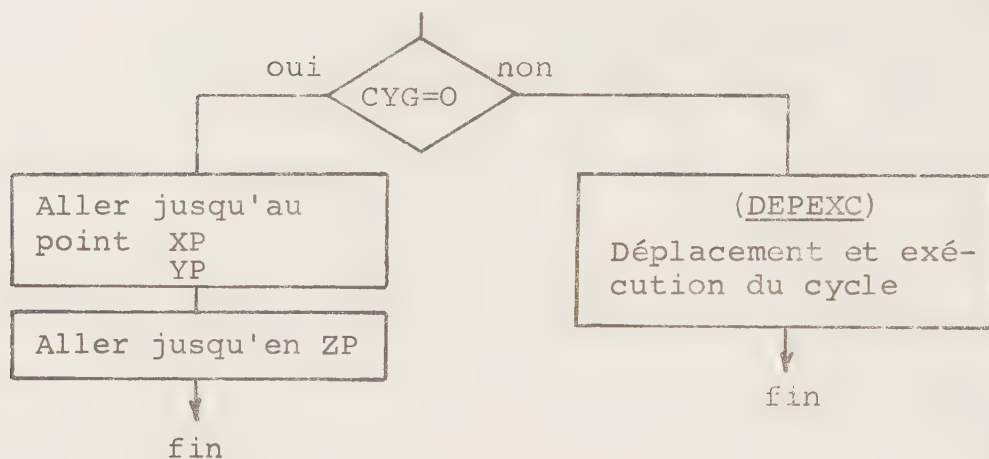
(CHOIX)

fin

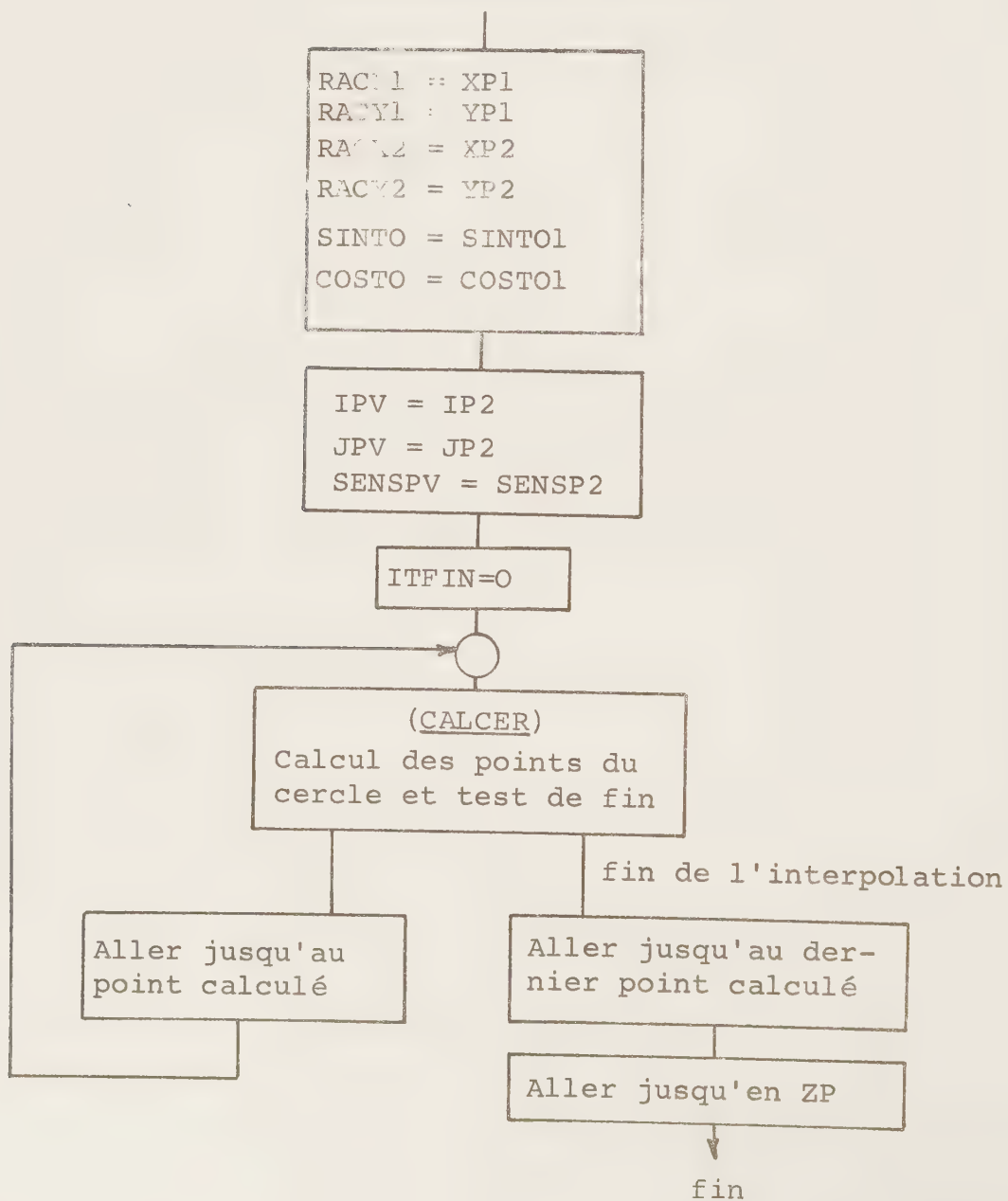


EXDRT

(Exécution d'une droite)

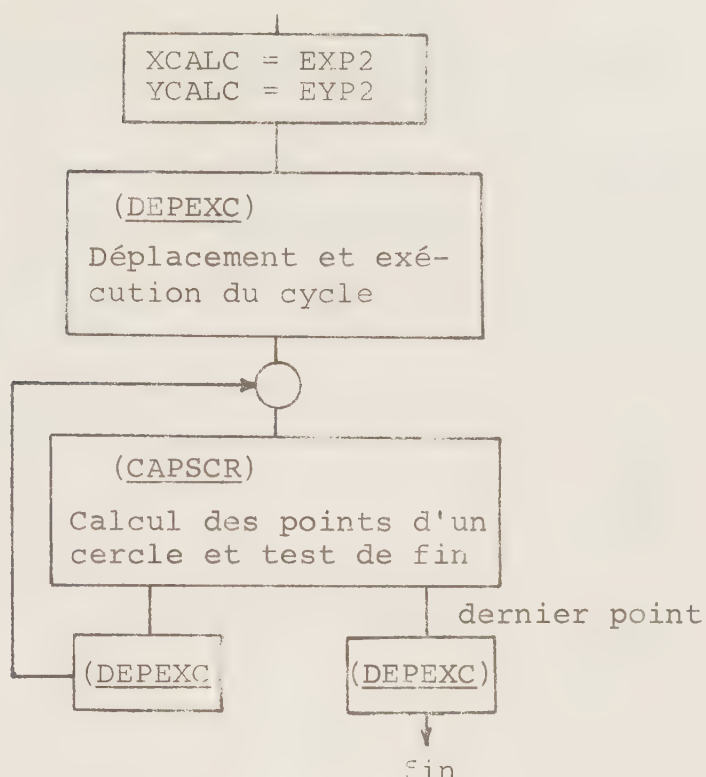
EXCER

(Exécution d'un cercle)

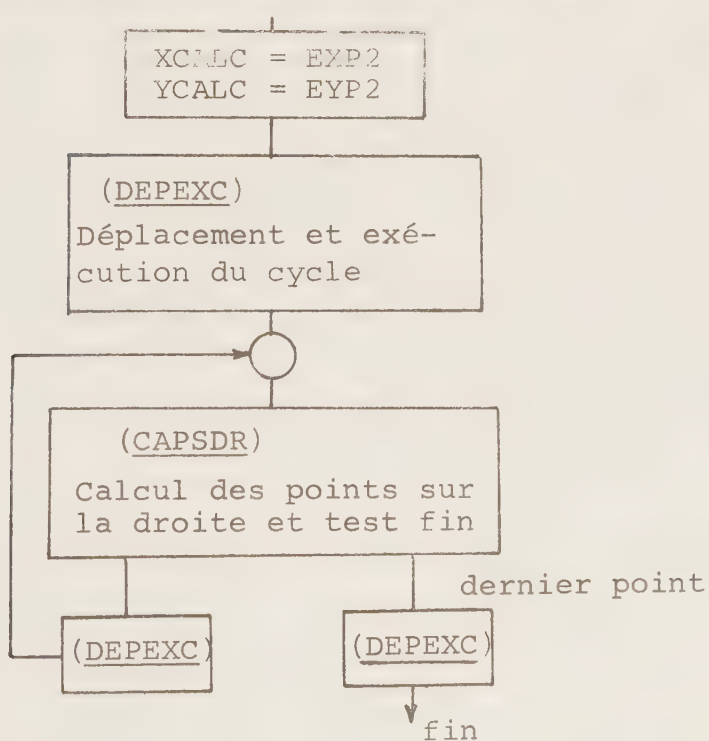


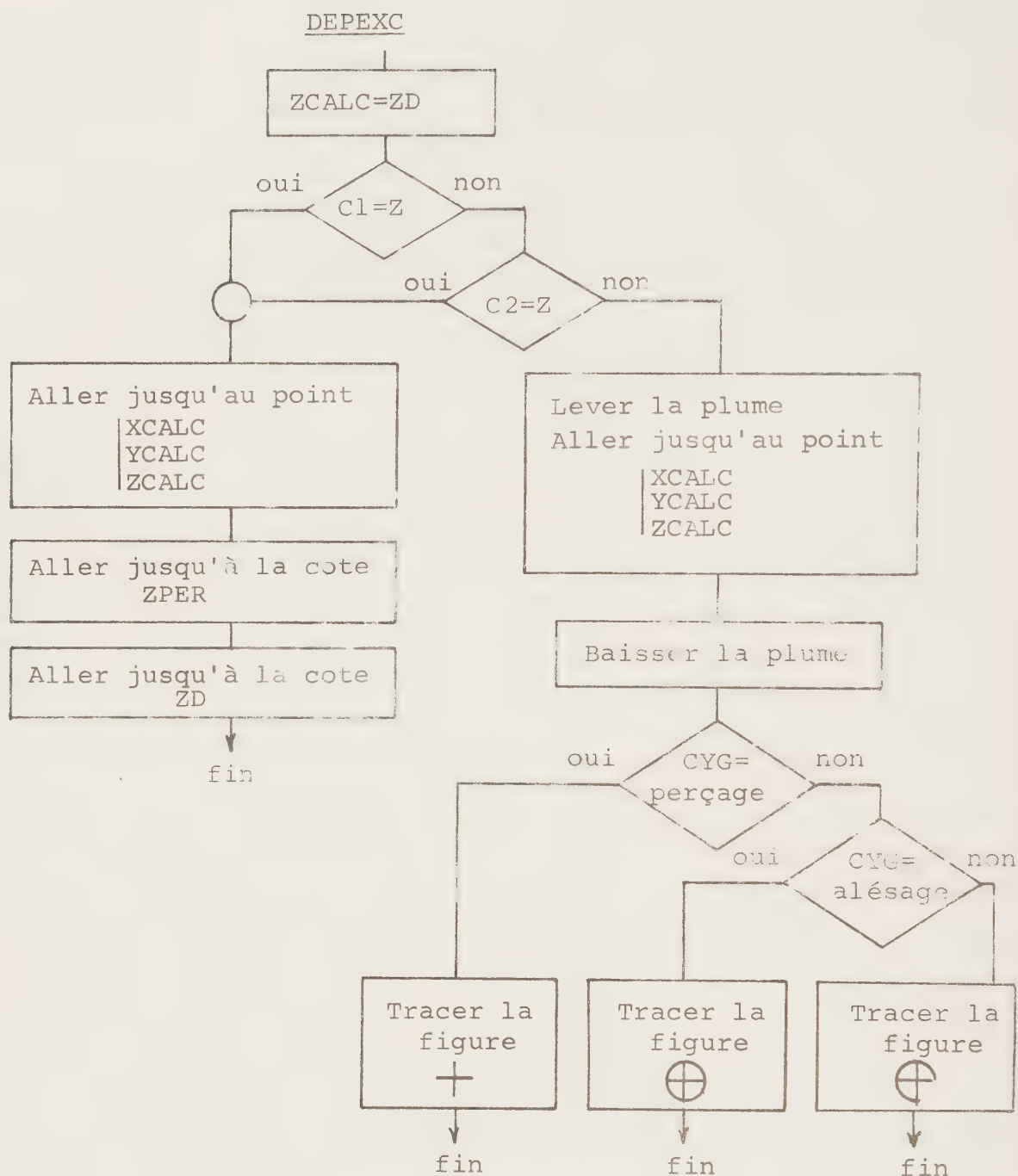
EXPCE

(Exécution d'une paterne cercle)

EXPDR

(Exécution d'une paterne droite)





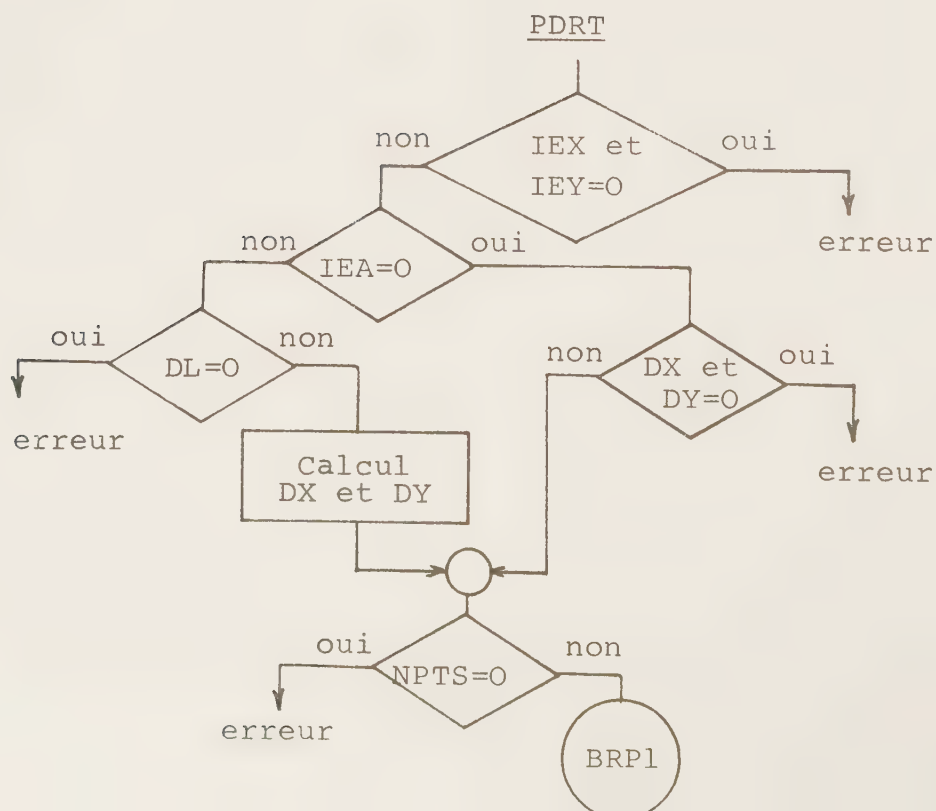
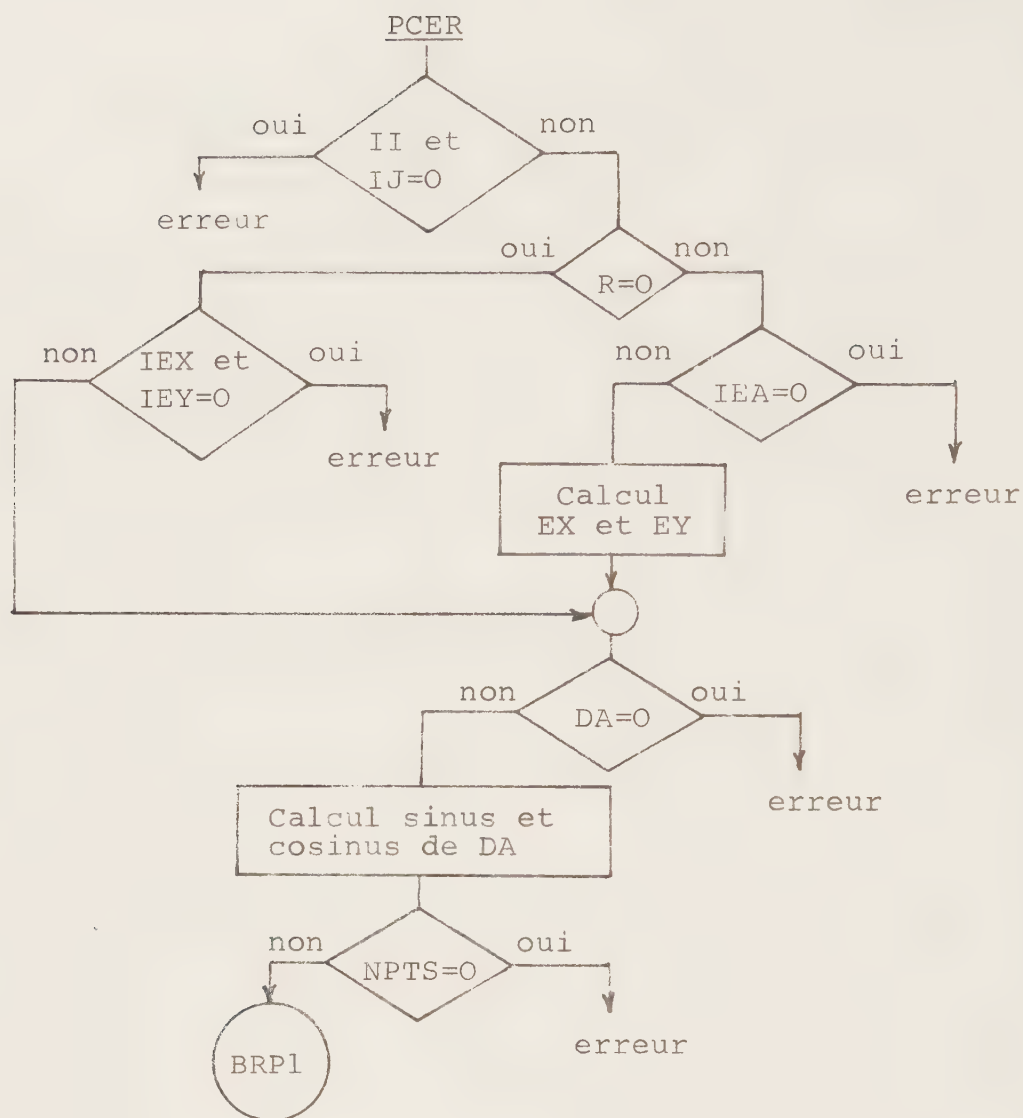
VI - Programmes de traitement des fonctions lues.

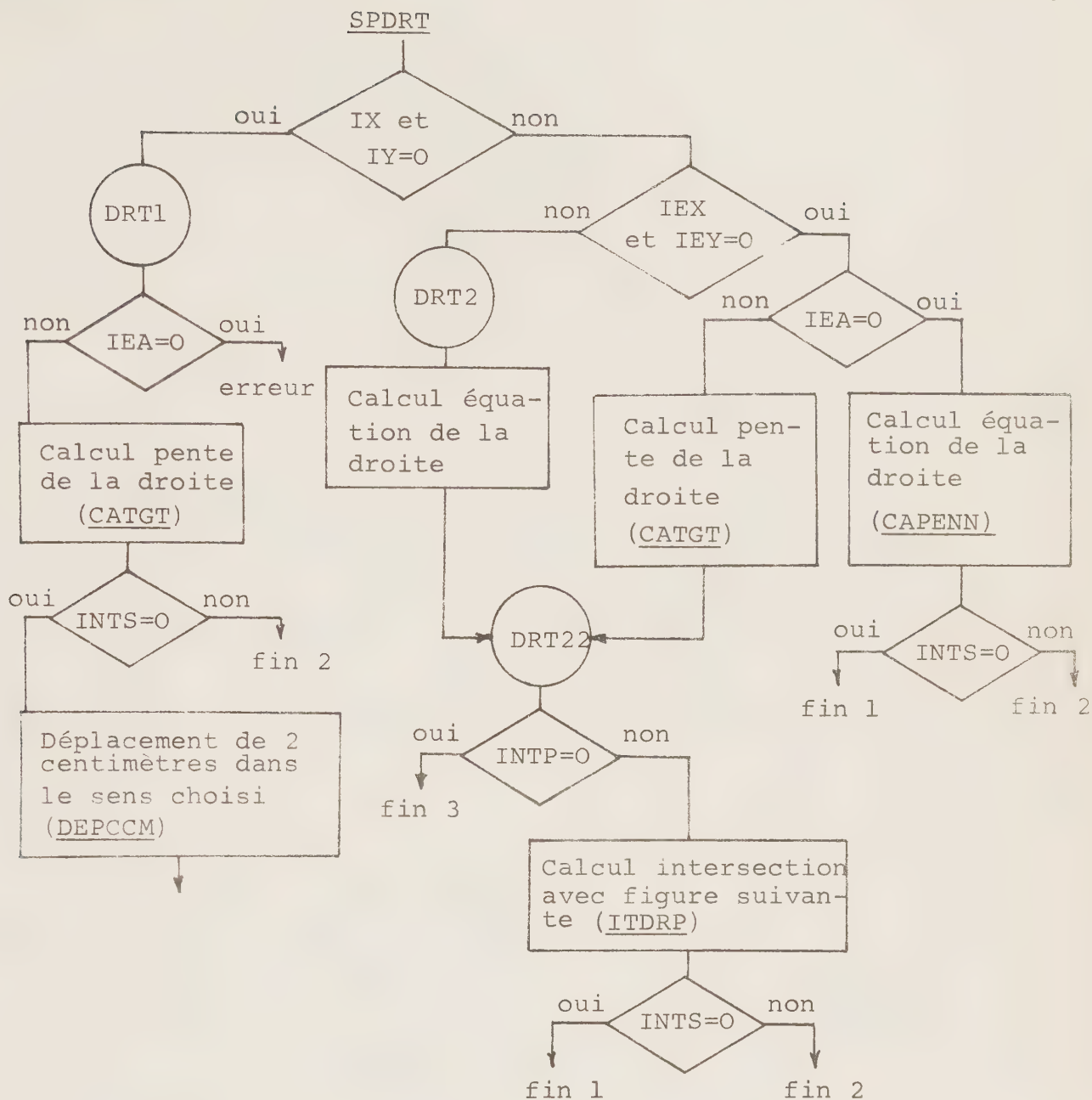
Ces sous-programmes assurent le traitement des diverses fonctions qui peuvent apparaître. Ils examinent les informations du bloc de définition géométrique et détectent celles qui sont contradictoires ou manquantes. S'il n'y a aucune erreur, ils assurent les appels des programmes de calculs géométriques nécessaires pour trouver la forme canonique de la fonction.

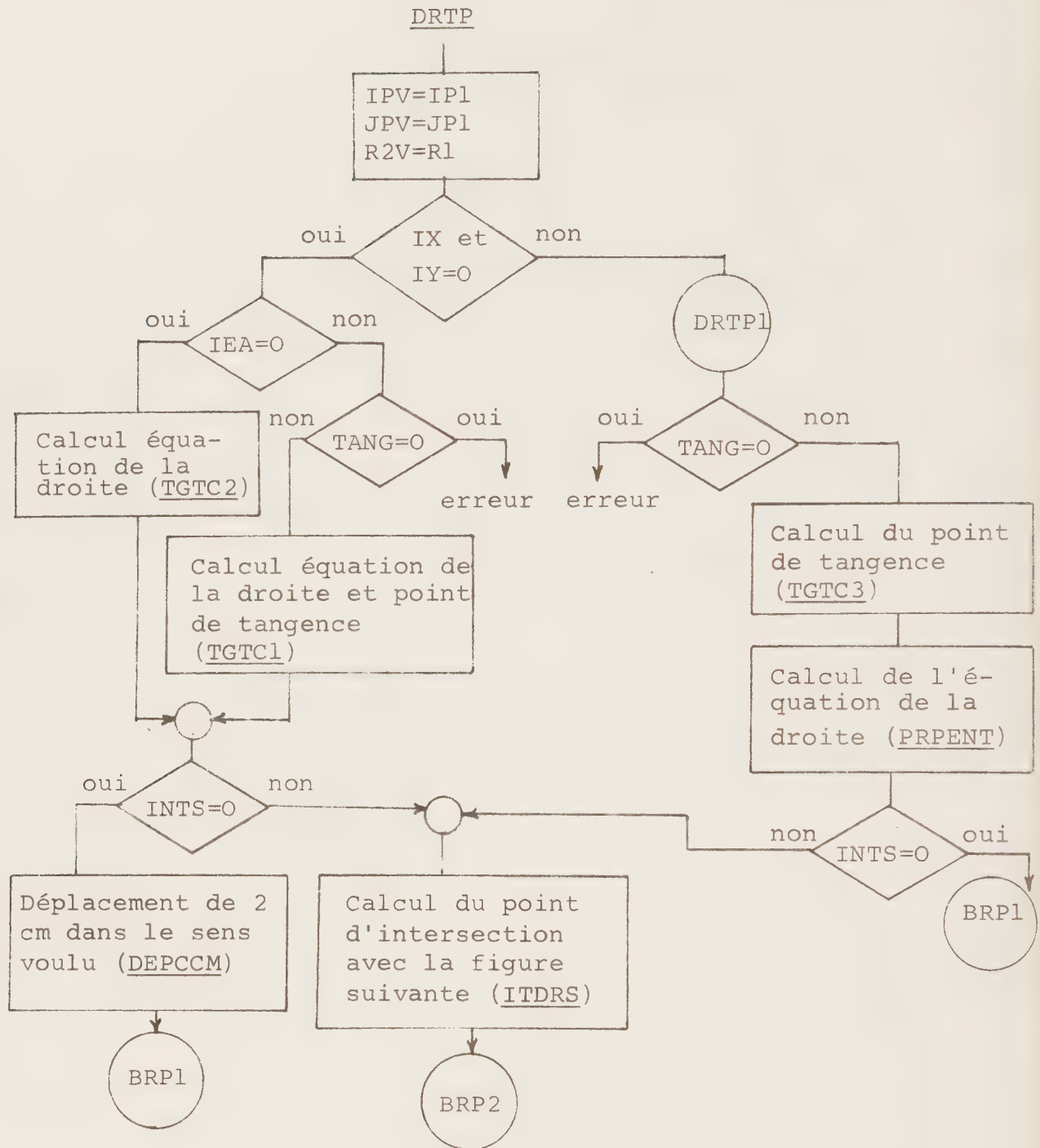
Les organigrammes correspondant explicitent succinctement leur déroulement qu'il serait trop long de décrire.

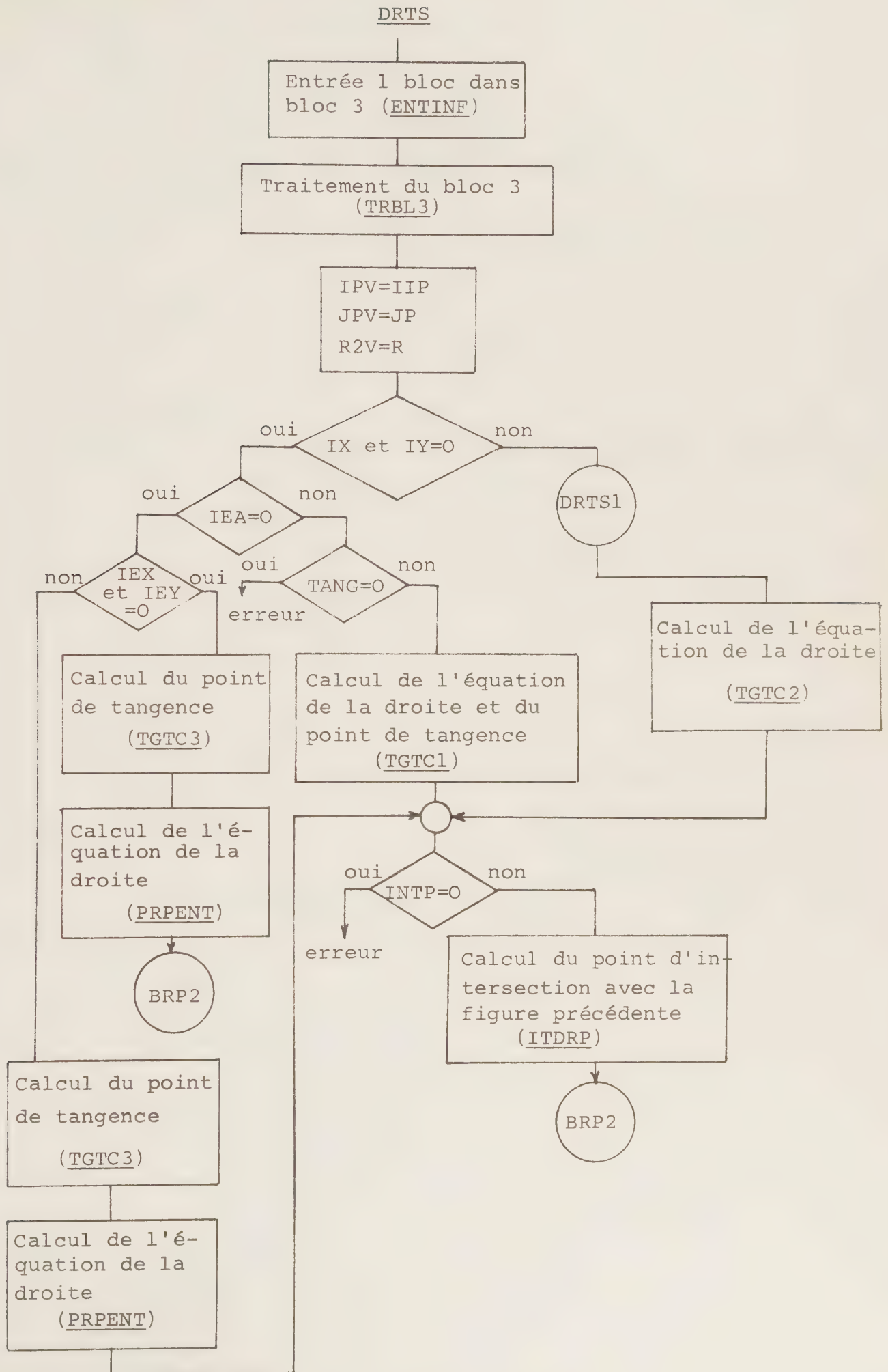
Notons toutefois que trois sorties distinctes se présentent

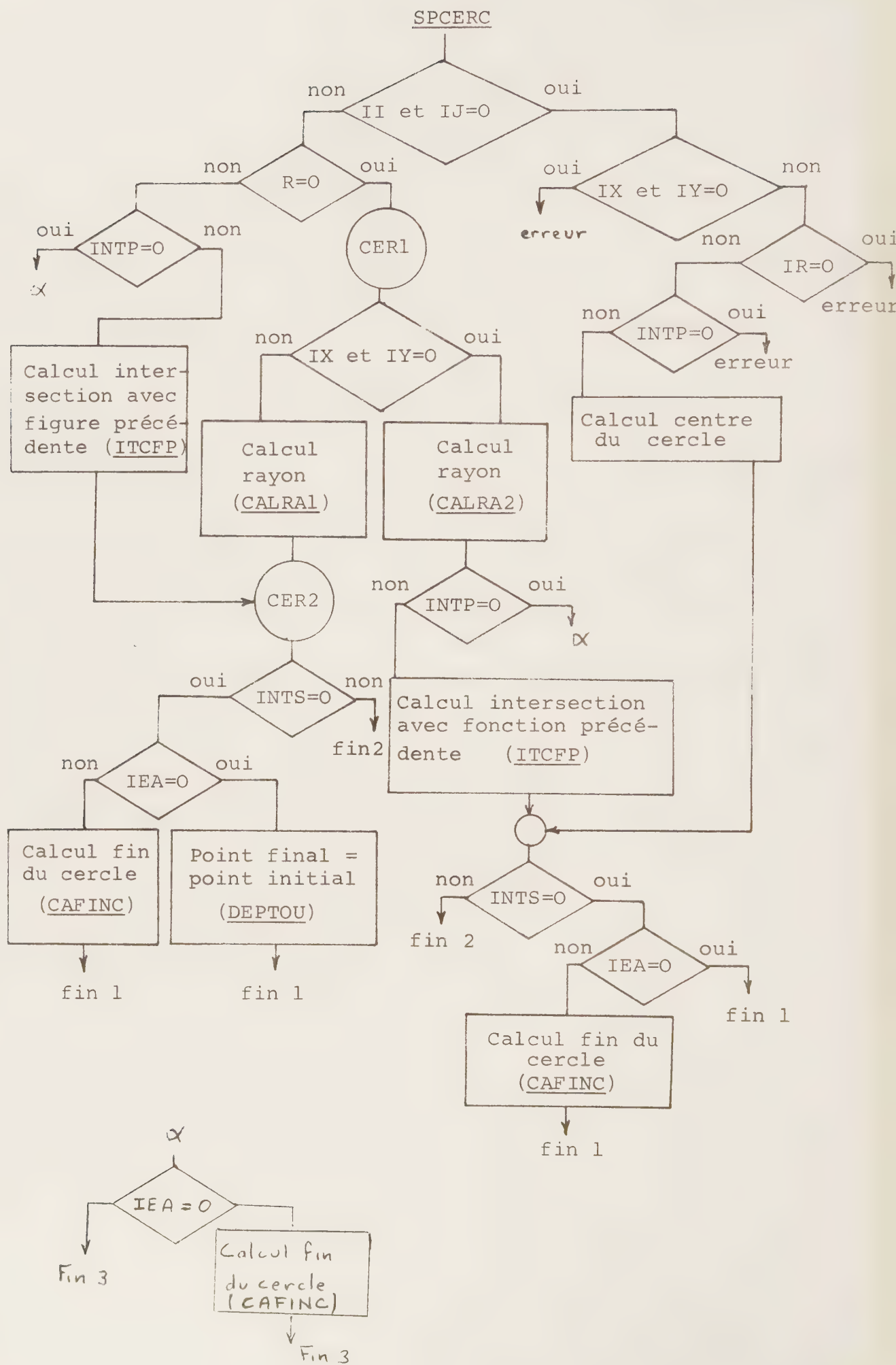
- la sortie erreur, s'il y a une faute de définition
- la sortie BRP1 qui correspond à une fonction définie par un seul bloc d'informations
- la sortie BRP2 qui correspond à une fonction faisant appel à deux blocs d'informations.

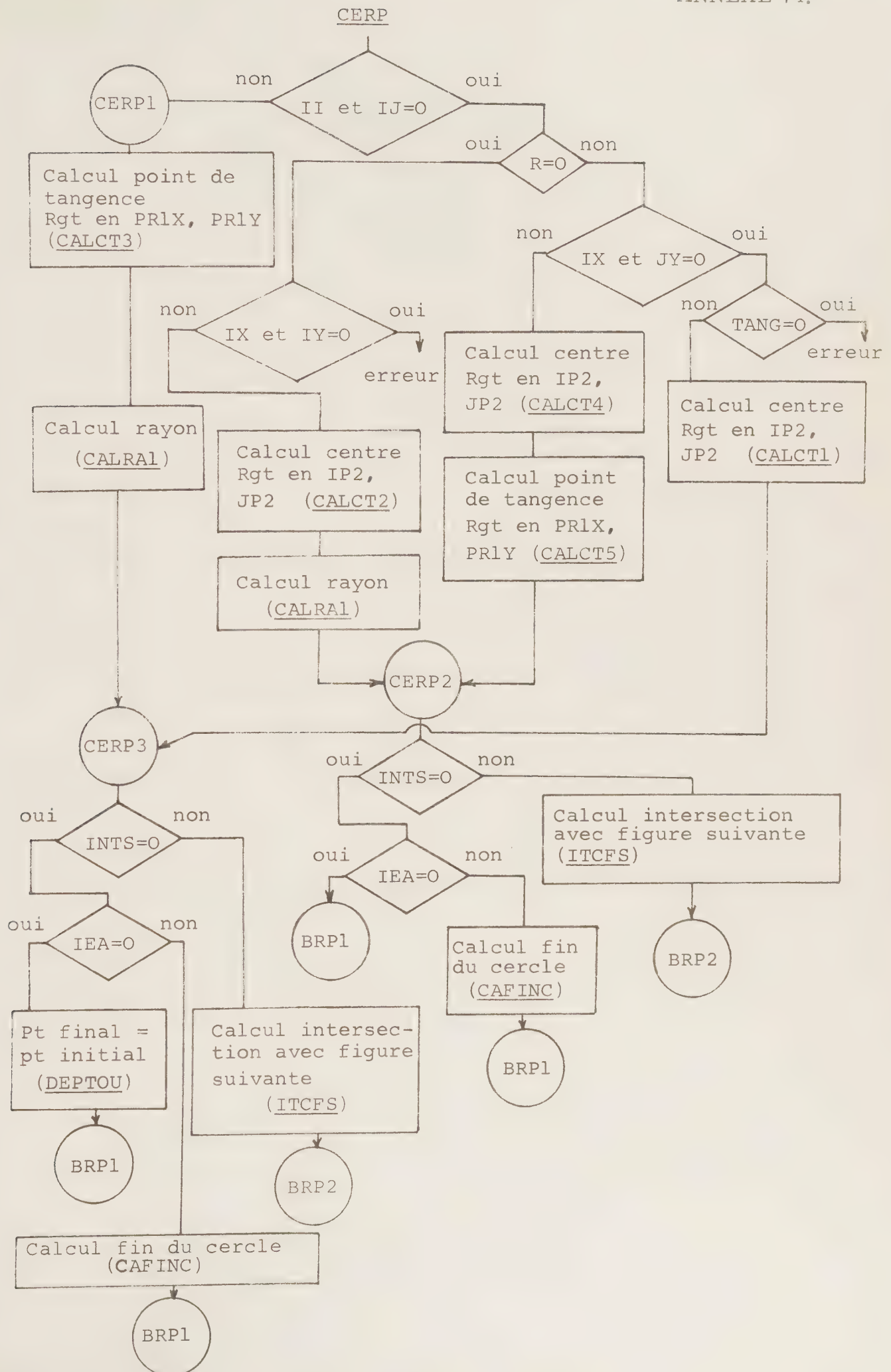


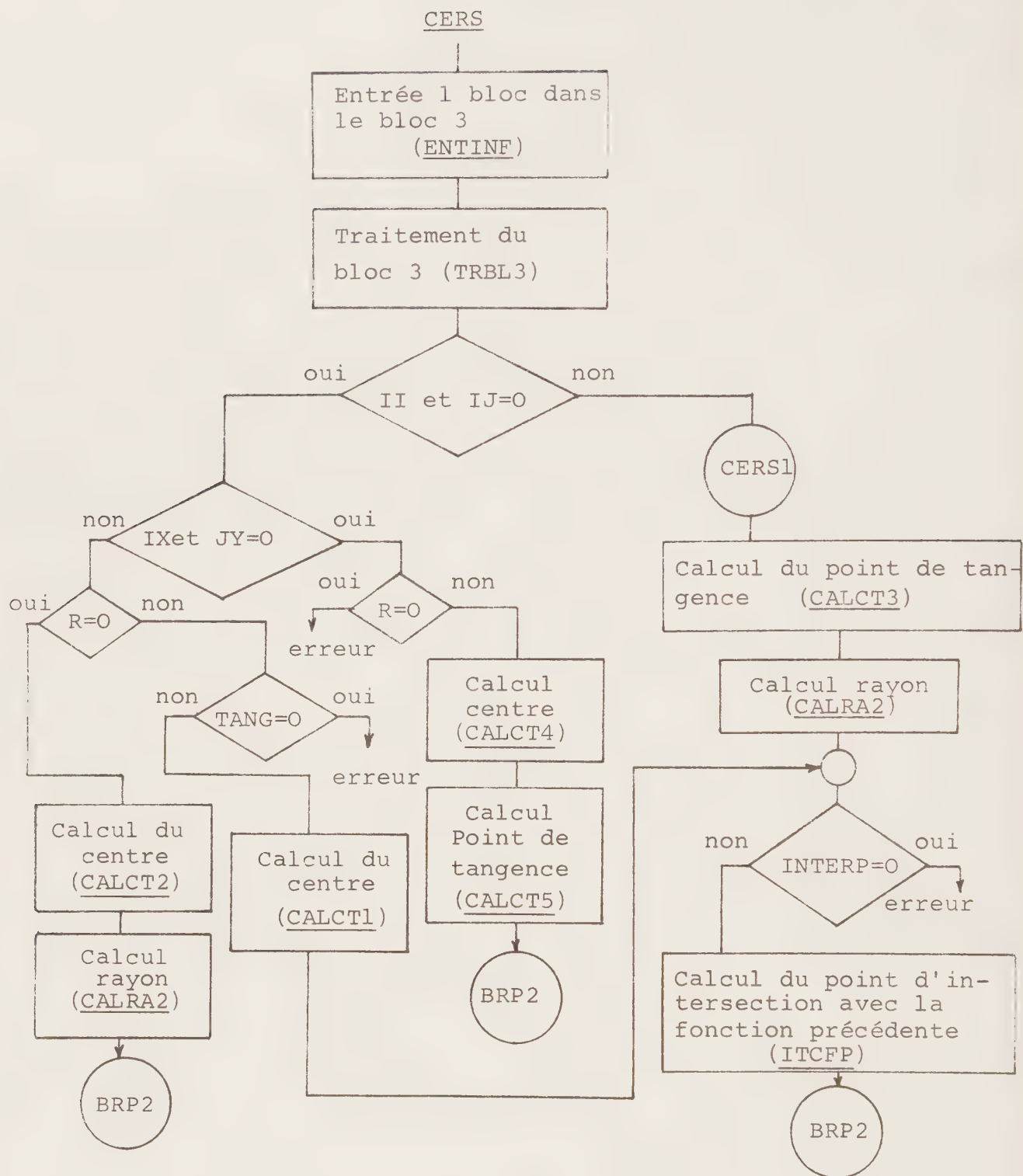


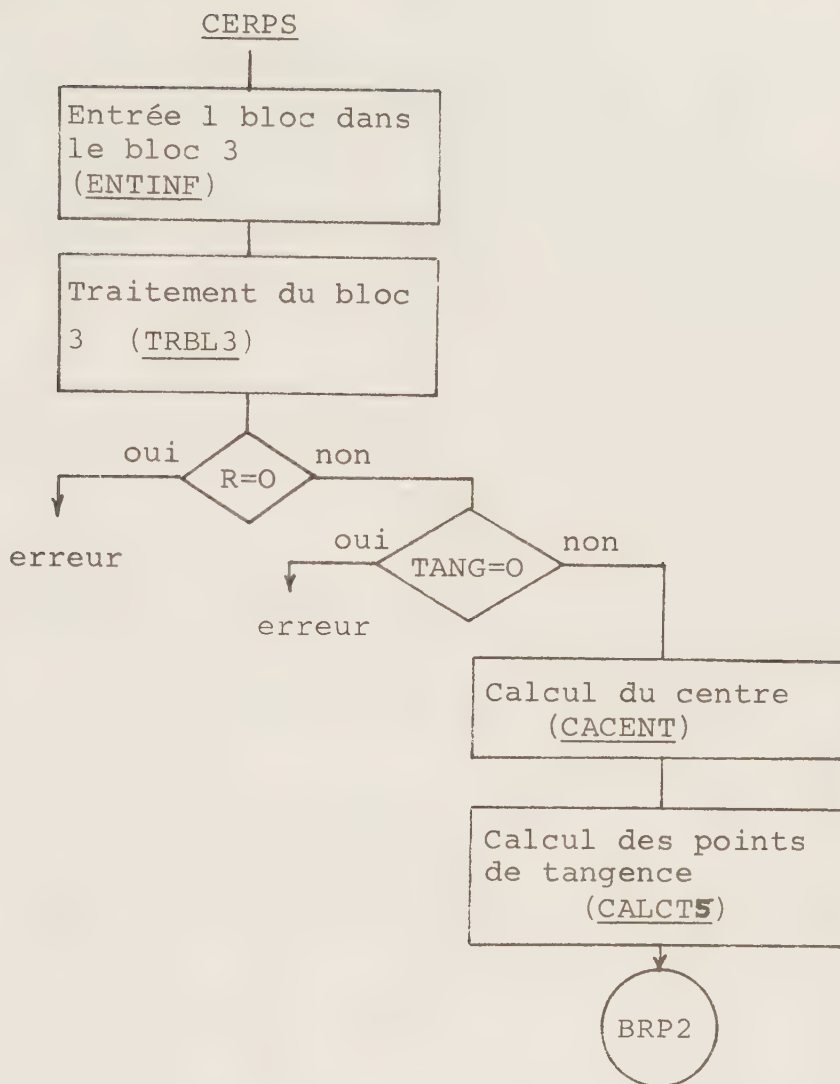












VII - Sous-programmes de sortie des informations symboliques
 (SORBL1) ou "machine" (SRBL1M).

Ces deux sous-programmes sont utilisés respectivement par les processeurs GBS et GBM.

Le premier (SORBL1) restitue sous forme symbolique les informations géométriques programmées. Les fonctions apparaissent alors sous la forme canonique suivante :

- pour la droite : coordonnées du point final
- pour le cercle : coordonnées du centre et du point final.

Le second sous-programme (SRBL1M) est plus complexe puisqu'il doit fournir des informations acceptées par l'armoire de commande de la machine outil et ce, compte-tenu du rayon et de la longueur de l'outil programmés.

Dans une première partie, le programme SRBL1M détermine les fonctions préparatoires à perforer (vitesse avance, de rotation, arrosage, etc...).

Puis, compte-tenu du côté d'usinage, du rayon de l'outil et des profondeurs de passe programmés on effectue le calcul de la courbe parallèle à la courbe définissant le profil fini de la pièce.

Les points finaux correspondant à chaque bloc d'informations sont déterminés de la manière suivante. S'il s'agit du dernier bloc, le point final sera situé sur la normale à la figure définie dans ce bloc.

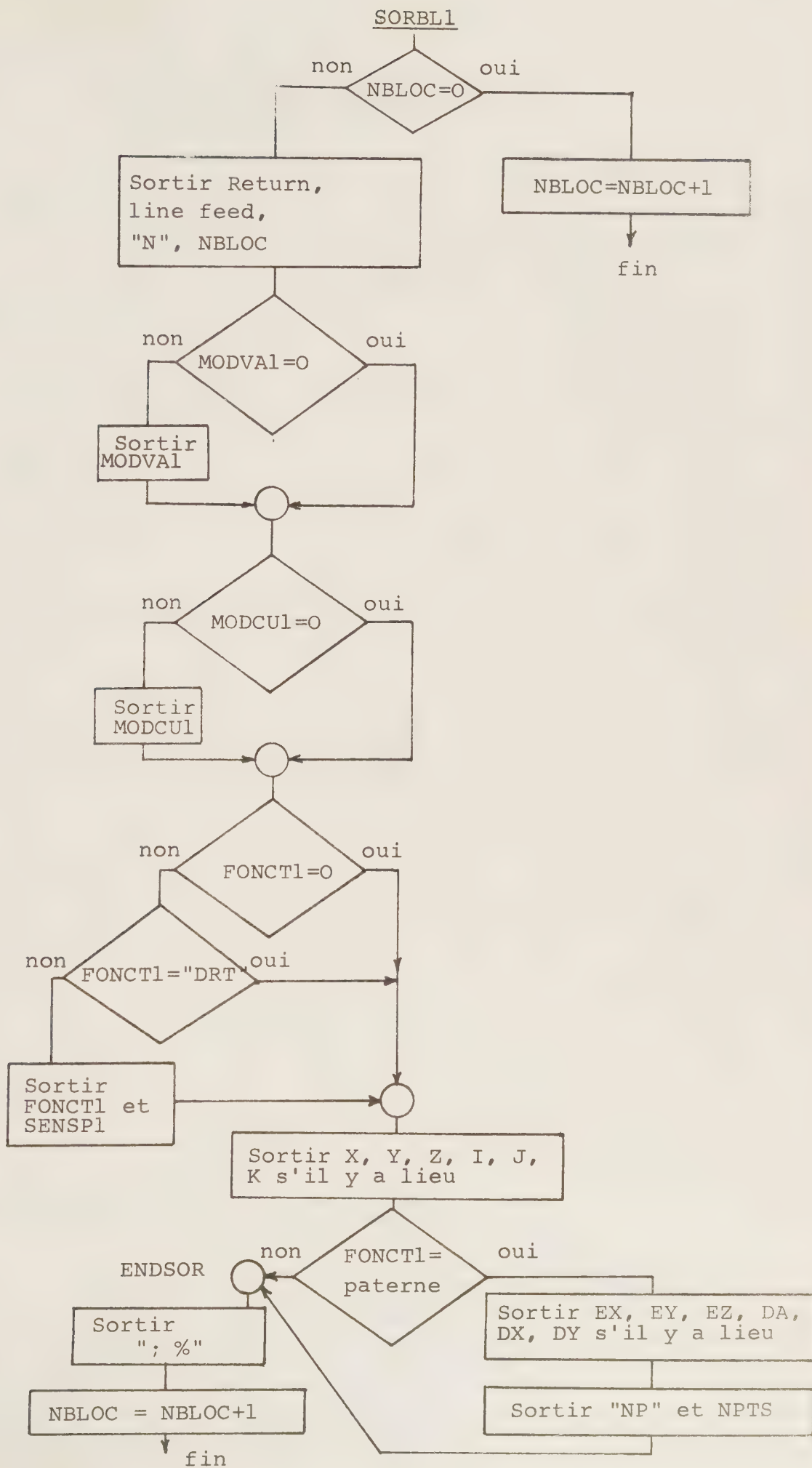
S'il s'agit du premier bloc ce point se trouvera sur la normale à la courbe décrite par le bloc suivant.

Dans les autres cas deux solutions peuvent se présenter :

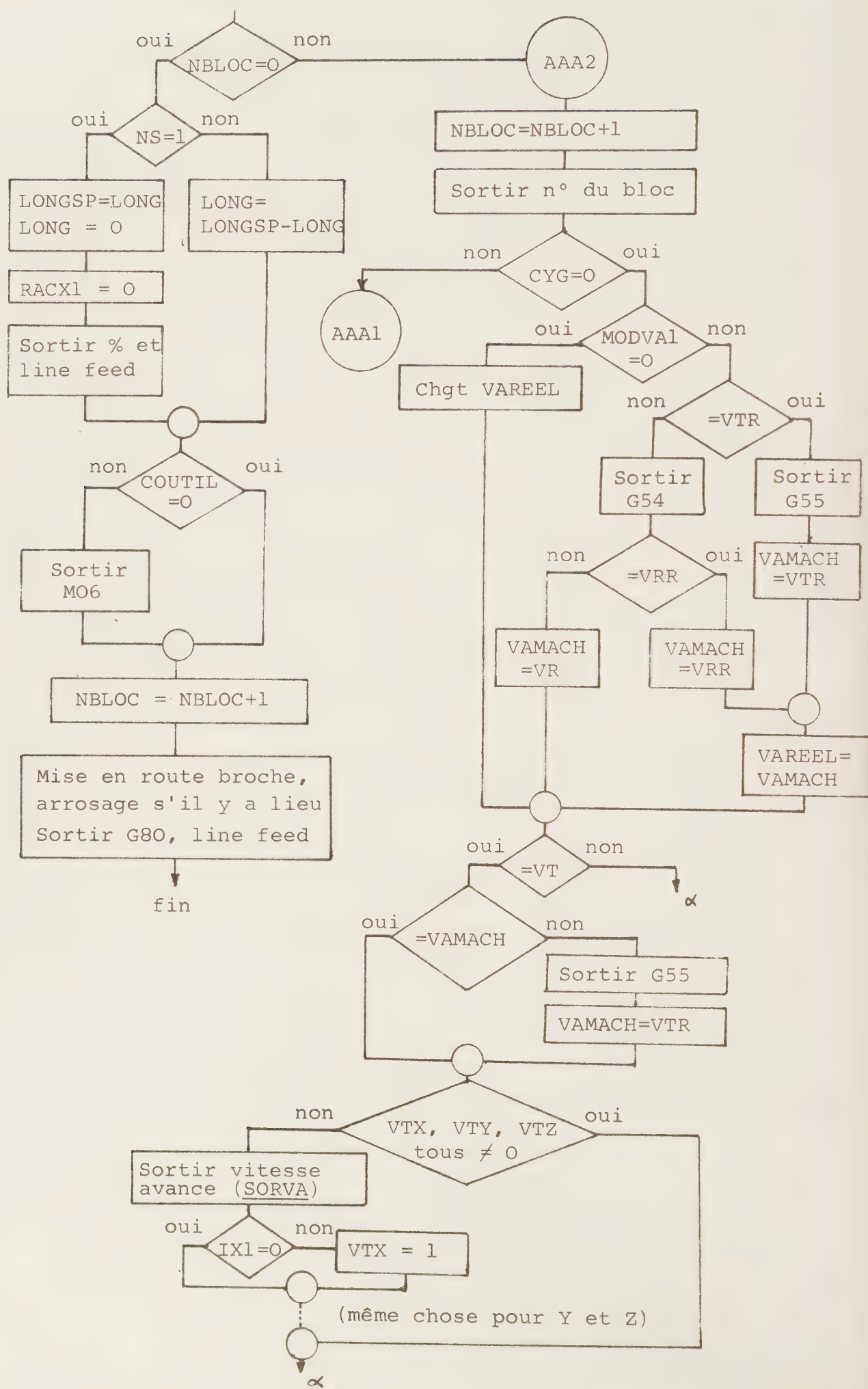
- les courbes parallèles admettent un point d'intersection qui sera le point final,

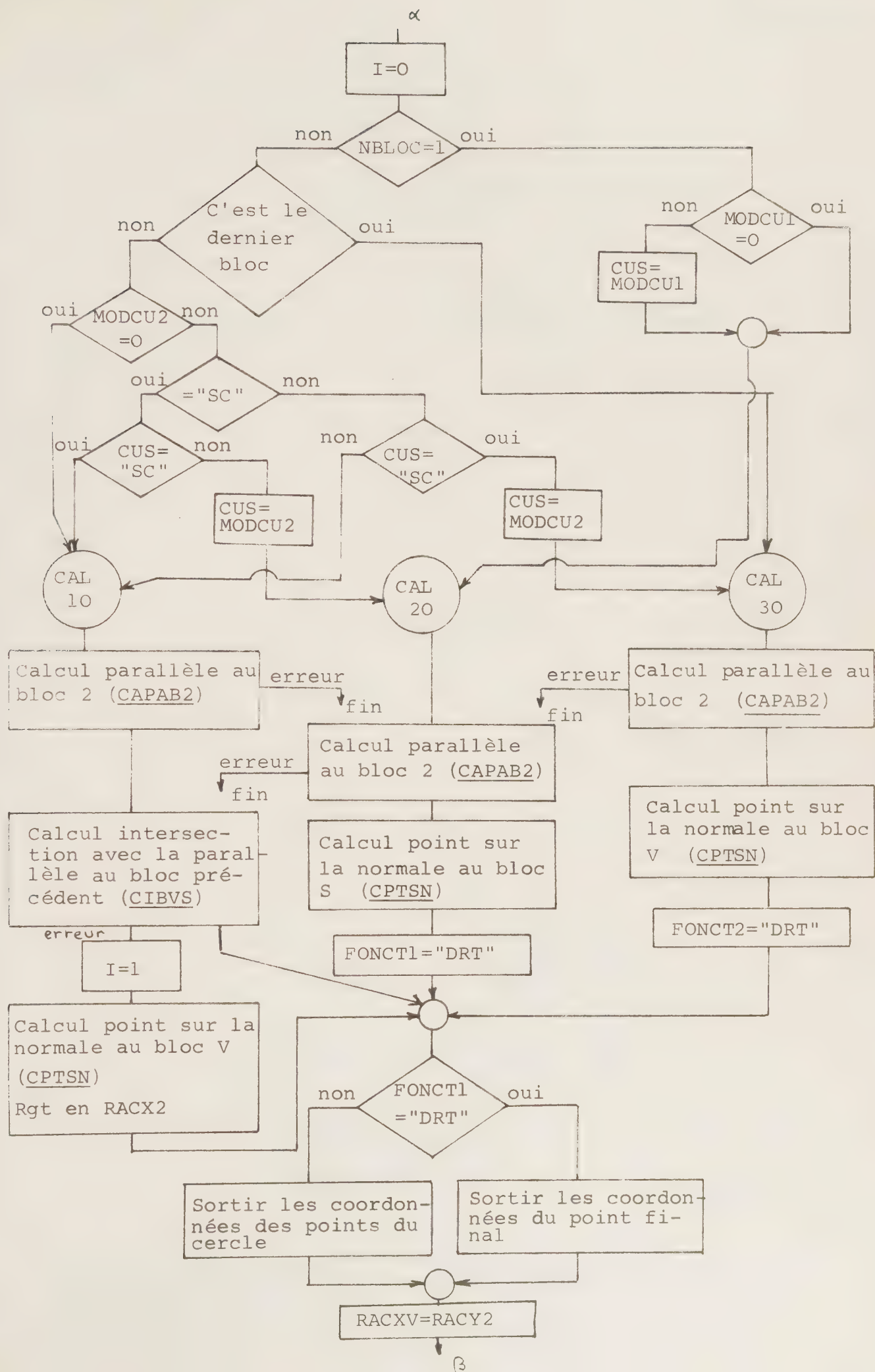
- les courbes parallèles n'admettent pas de point d'intersection. On passera alors d'une courbe à l'autre par une portion de cercle débutant sur la normale au premier bloc et aboutissant à la normale du second.

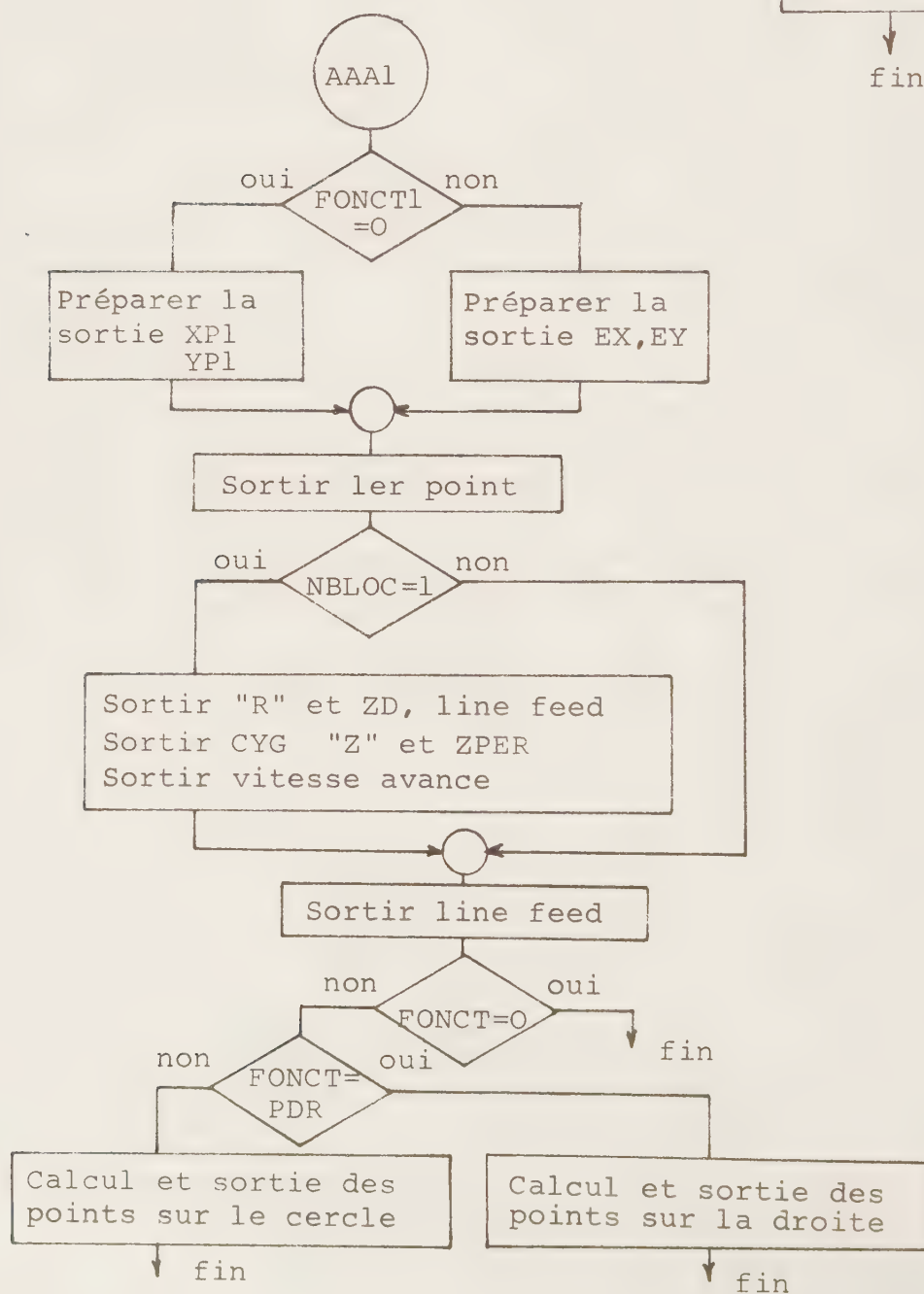
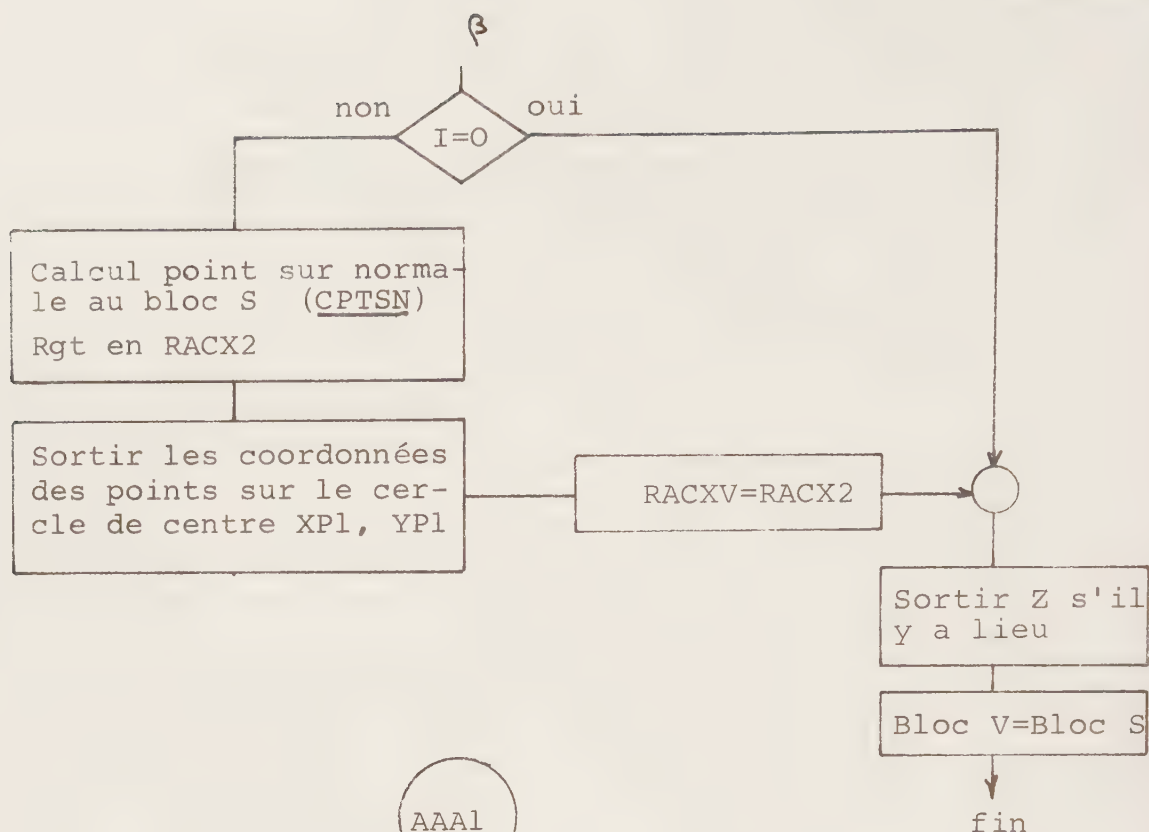
Notons que si dans un groupe d'informations apparaît la cote Z, le déplacement suivant cet axe sera effectué à la fin de la description de la courbe définie dans le bloc.

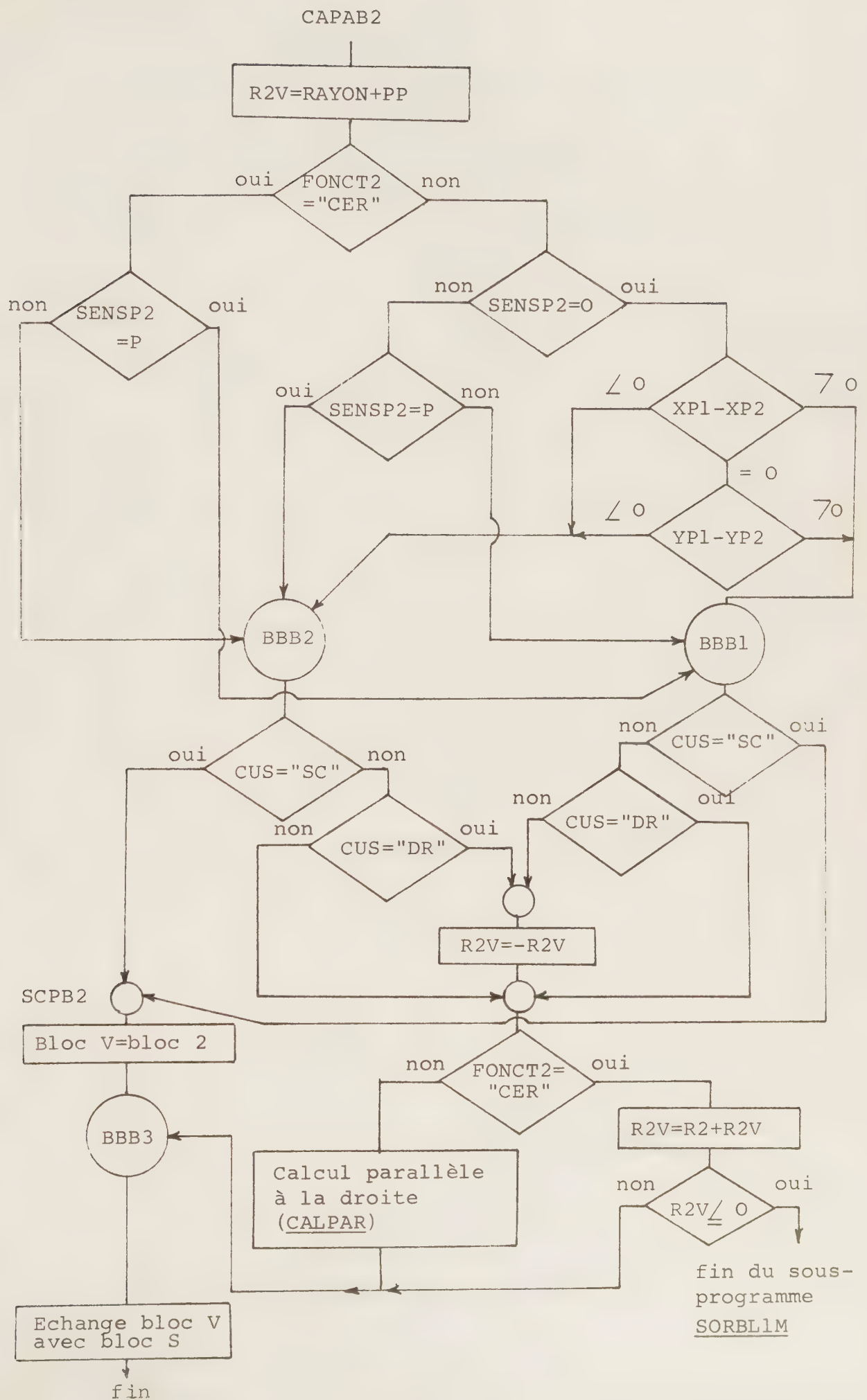


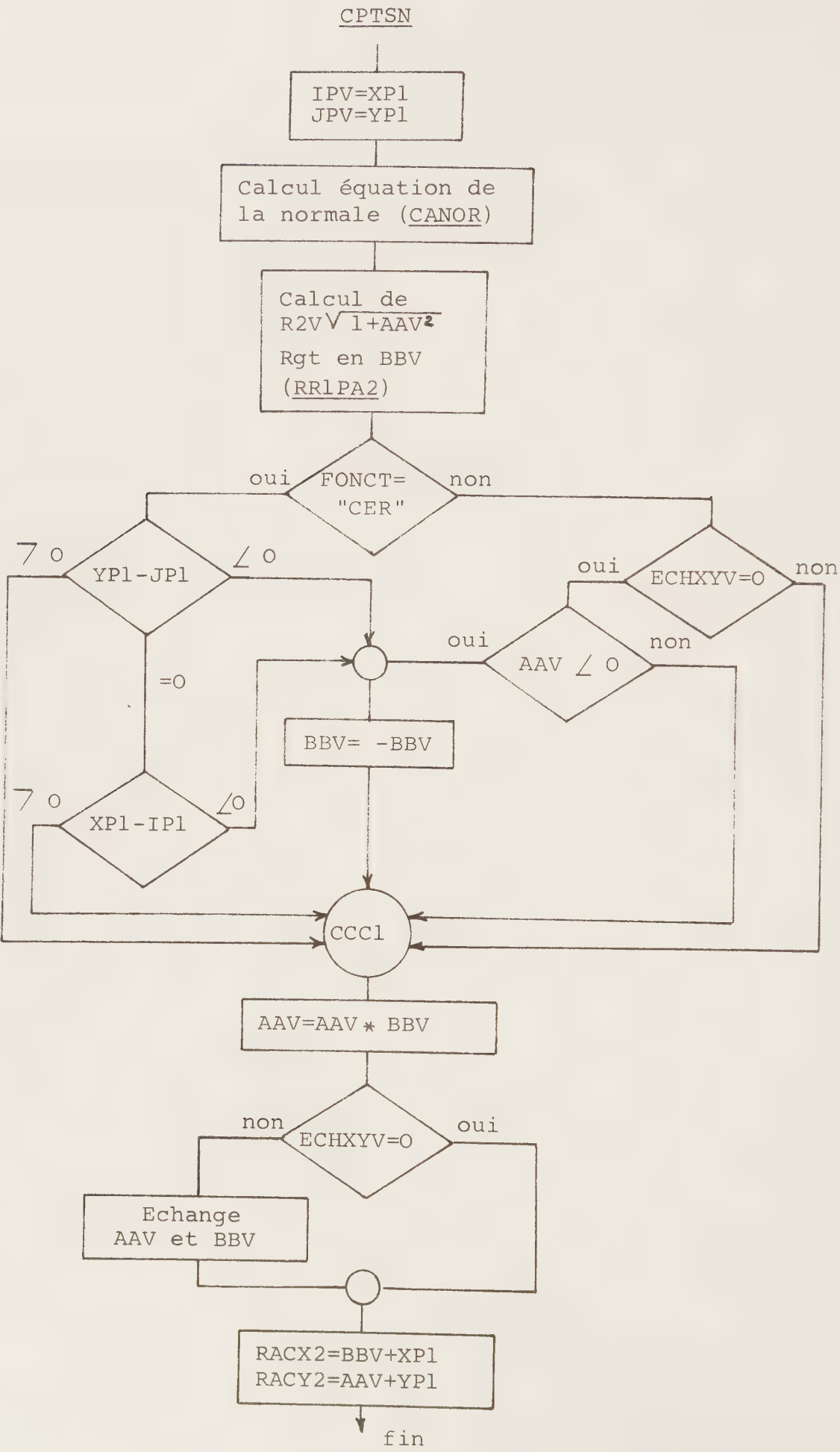
SRBL1M

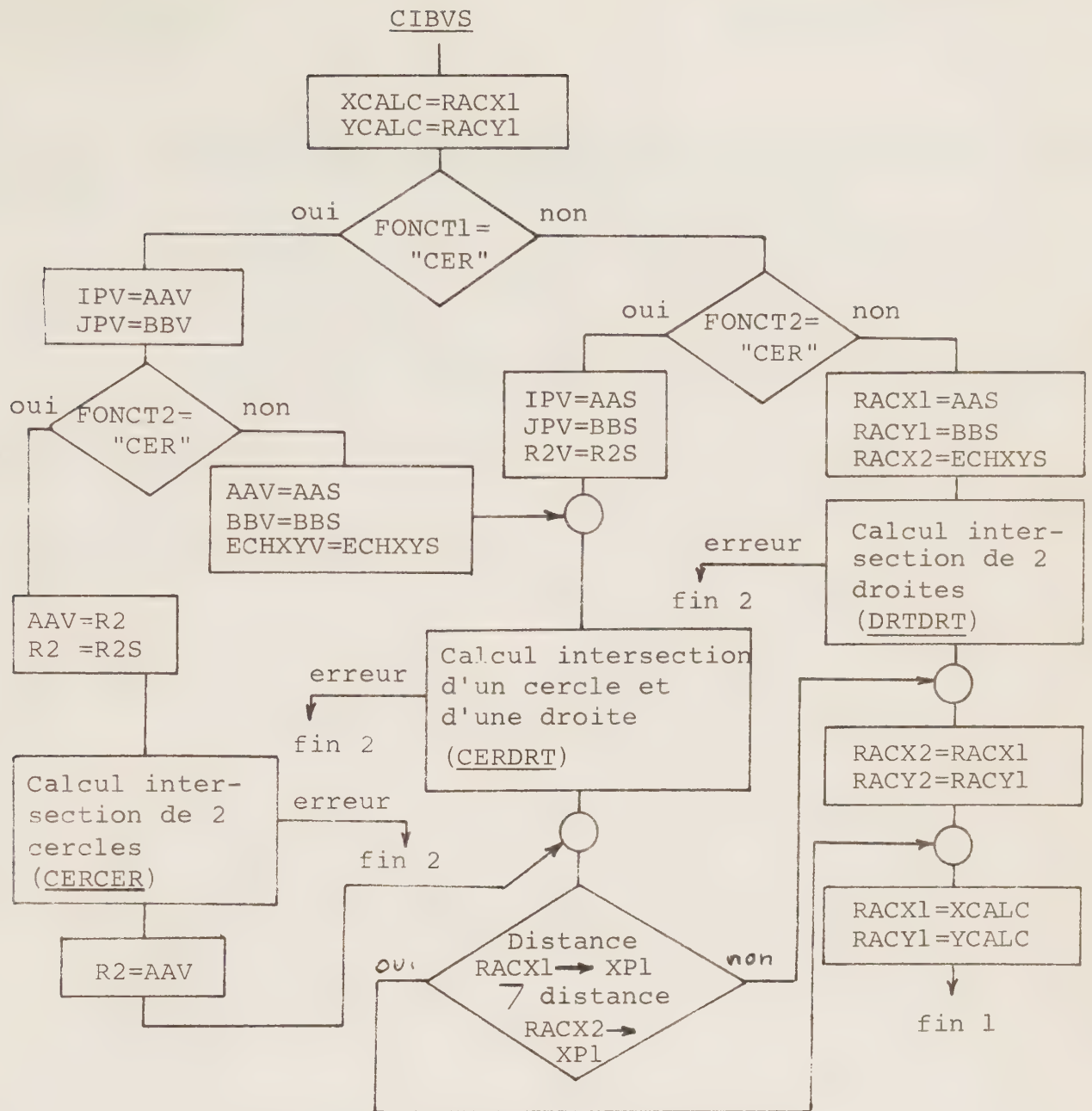












La partie du programme SRBL1M débutant en AA1 concerne la sortie des informations "machine" relatives à un cycle d'usinage.

D'autre part SRBL1M utilise plusieurs sous-programmes de calculs géométriques qui permettent de déterminer :

- l'équation de la parallèle à la fonction définie dans le bloc (CAPAI
- les coordonnées du point sur la normale à la courbe distant d'une valeur D (CPTSN)
- les coordonnées du point de concours des deux courbes parallèles aux figures programmées (CIBVS).

VIII - Sous-programmes divers.

VIII-1 SPTR (sous-programme de transcodage) :

Ce programme assure la recherche dans un premier tableau d'une valeur présente dans l'accumulateur.

Lorsqu'il l'a trouvée, il charge l'accumulateur d'une grandeur prise dans un tableau correspondant au premier.

En cas d'échec dans cette recherche, la sortie du sous-programme diffère et le retour se fait à l'instruction suivant l'appel du sous-programme

IRV SPTR

_____	retour si erreur
_____	retour normal

VIII-2 SPSND (sous-programme de sortie d'un nombre décimal) :

Les nombres décimaux sont exprimés ici en virgule fixe double longueur, la virgule se situant entre les deux mots.

Le sous-programme SPSND assure la sortie de tels nombres en écrivant les zéros non significatifs. La séquence d'appel sera la suivante :

IRV	SPSND	
2007	004	format
X	XN	grandeur à sortir
		retour

Le format permet d'indiquer s'il s'agit d'un nombre en simple ou double longueur, et précise le nombre total de chiffres et le nombre de chiffres après la virgule, qui n'est d'ailleurs jamais imprimée (cela est imposé par le format accepté par l'armoire de commande).

VIII-3 SPLND (sous-programme de lecture d'un nombre décimal) :

Les nombres à traiter sont toujours exprimés en virgule fixe et double longueur.

La séquence d'appel reste la même que pour le sous-programme SPSND sauf pour le retour au programme principal, qui se fait en séquence s'il y a une erreur dans l'écriture du nombre, et avec saut si celui-ci est correct.

Exemple :

```

IRV SPLND
2007 004
X    XN
_____ retour si erreur
_____ retour normal

```

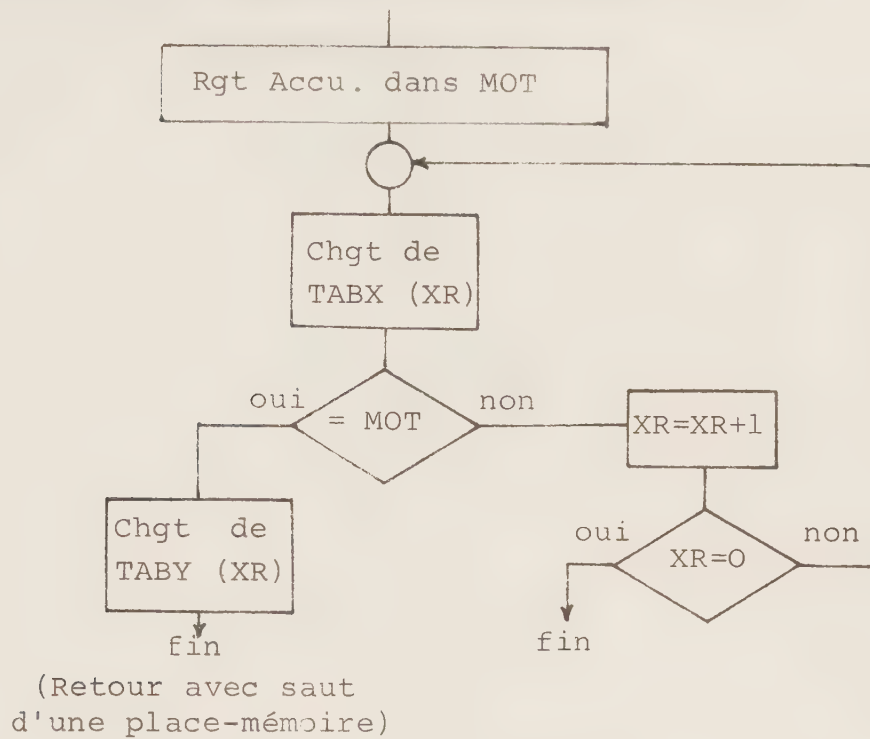
Notons qu'il ne faut jamais indiquer la virgule ce qui entraîne que les zéros après la virgule doivent être mentionnés. Par contre ceux qui la précèdent et ne sont pas significatifs peuvent être omis.

VIII-4 SPMESS : Sortie de message :

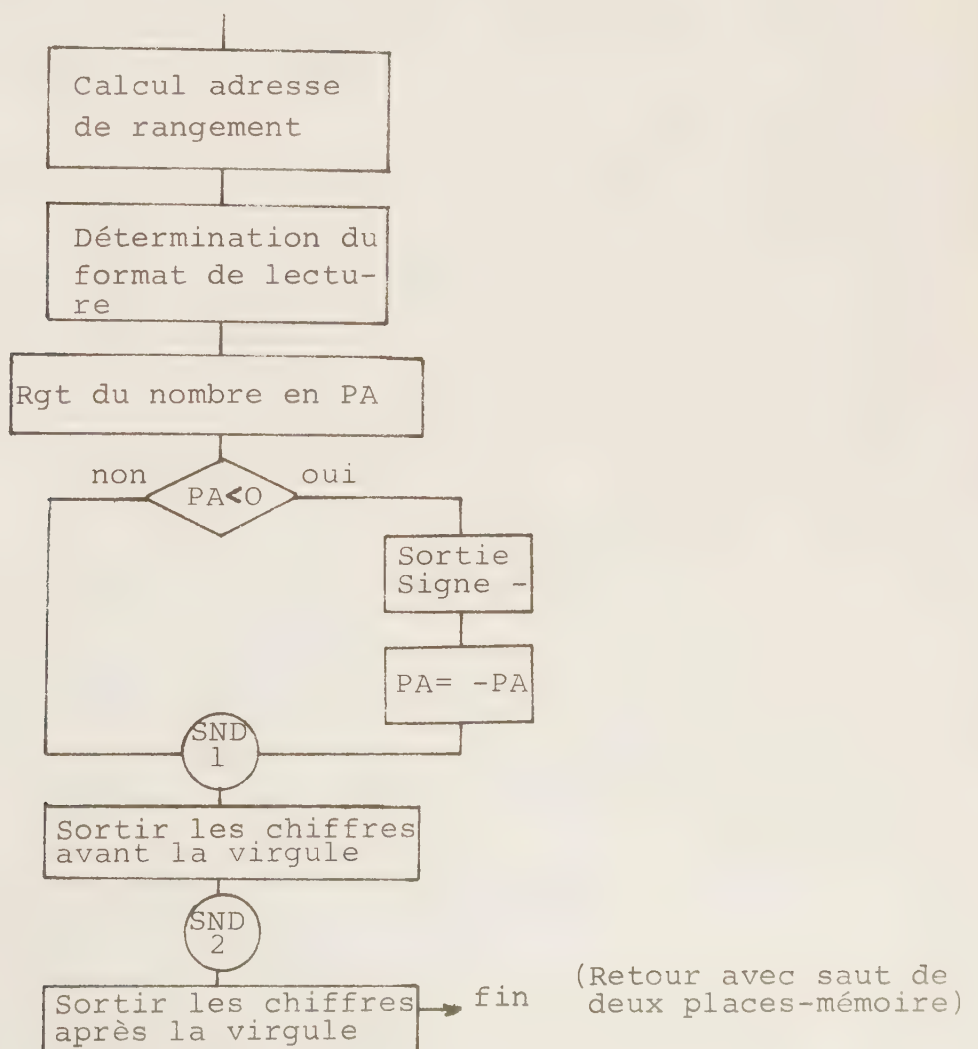
Ce sous-programme permet l'impression d'un message d'erreur rangé dans un tableau à raison de 3 caractères par mot. Le caractère @ indique la fin du message.

SPTR

(Sous-programme de transcodage)

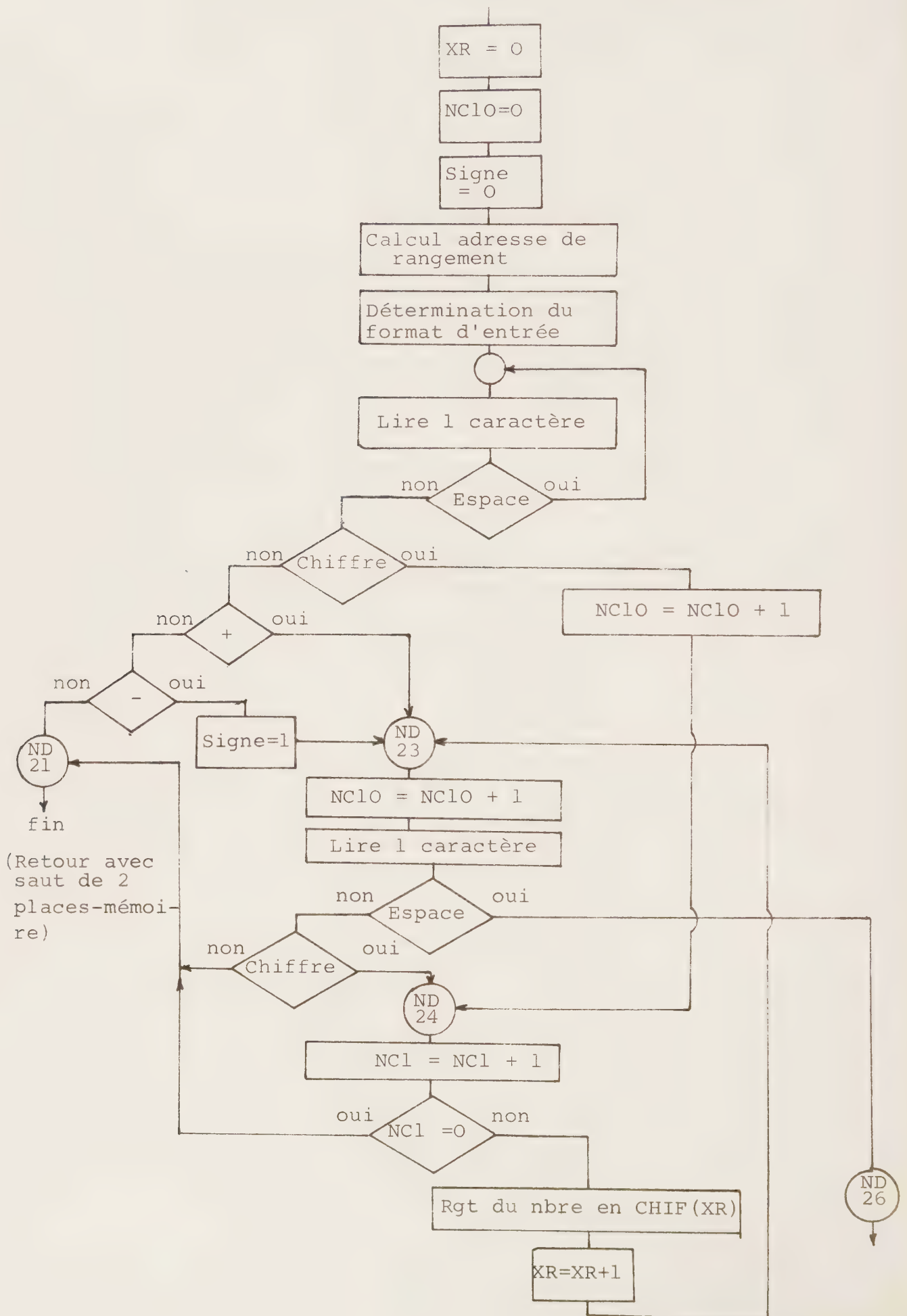
SPSND

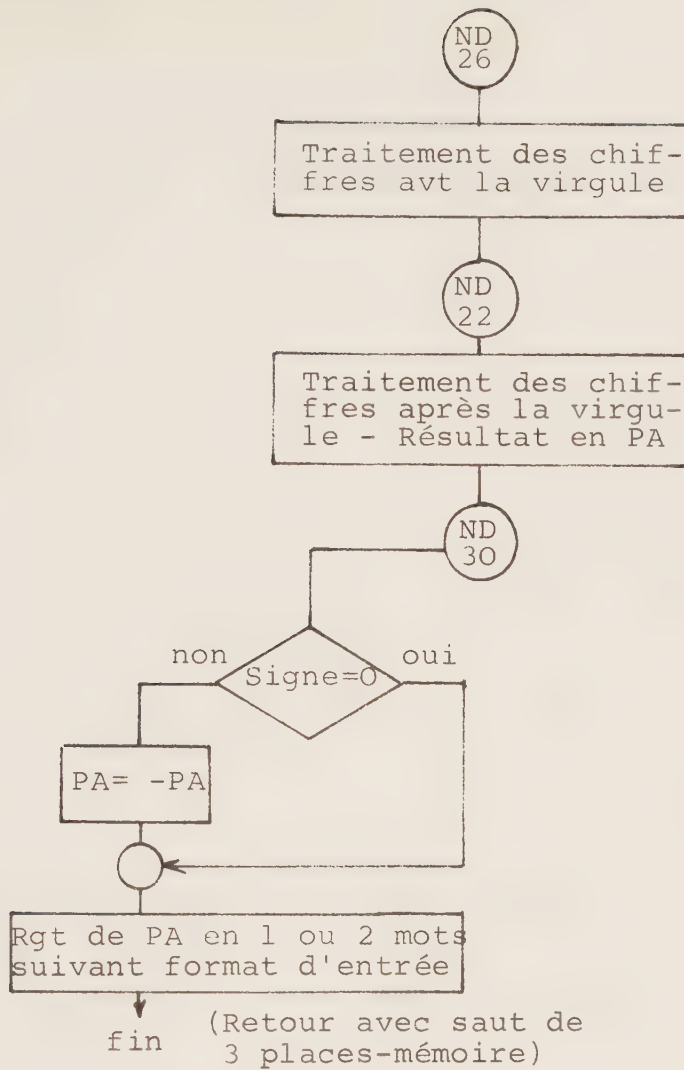
(Sous-programme de sortie d'un nombre décimal)



SPLND

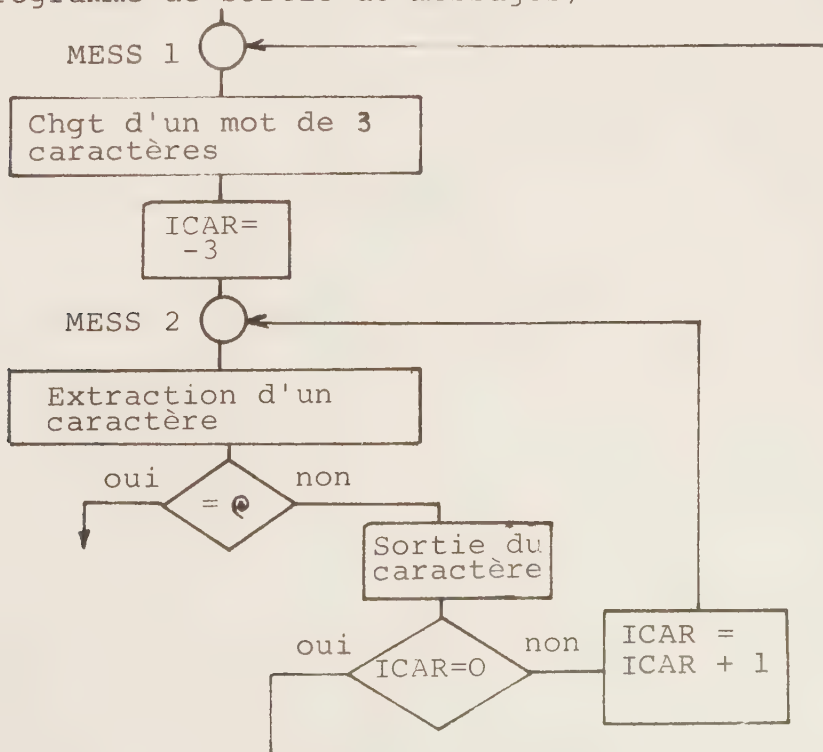
(Sous-programme de lecture d'un nombre décimal)





SPMESS

(sous-programme de sortie de messages)



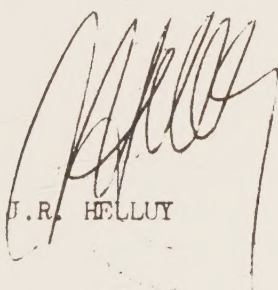
NOM DE L'ETUDIANT : MUSSE JEAN-PIERRE

Nature de la Thèse : DOCTEUR-INGÉNIEUR

Vu, Approuvé
et permis d'imprimer

NANCY, le 28 Février 1973 .

Le Président du Conseil de l'Université de NANCY I :



J.R. HELLUY

